

AD-A048 769

COMPUTER SCIENCES CORP SAN DIEGO CALIF

F/G 17/2

SIMULATION CODE FOR UNDERWATER TELEVISION SYSTEMS (SCOUTS). (U)

N00123-74-C-0312

SEP 77 A GORDON, T SELDEN

UNCLASSIFIED

NOSC-TR-146

NL

1 of 1
AD
A048769



END
DATE
FILED
2- 78
DDC

12^bs
12769

NOSC TR 146 AD A 048769

NOSC

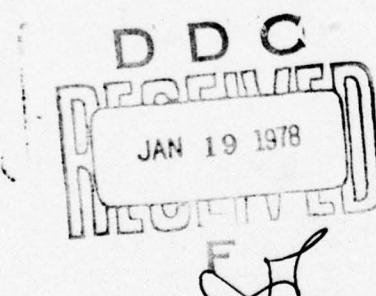
NOSC TR 146

Technical Report 146

SIMULATION CODE FOR UNDERWATER TELEVISION SYSTEMS (SCOUTS)

A Gordon
Naval Ocean Systems Center
T Selden
Computer Sciences Corporation
1 September 1977

Interim Report: June 1976 — May 1977
Prepared For
Naval Sea Systems Command



Approved for public release; distribution unlimited.

NAVAL OCEAN SYSTEMS CENTER
SAN DIEGO, CALIFORNIA 92152

W M
JDC FILE COPY



NAVAL OCEAN SYSTEMS CENTER, SAN DIEGO, CA 92152

AN ACTIVITY OF THE NAVAL MATERIAL COMMAND

RR GAVAZZI, CAPT USN

Commander

HL BLOOD

Technical Director

ADMINISTRATIVE INFORMATION

Work on this report was performed from 1 June 1976 to 1 May 1977. The work was funded as part of the Deep Ocean Technology Project by the Naval Sea Systems Command under Program Element 63713N. This report was reviewed for technical accuracy by Dr. Peter McCardell of the Naval Ocean Systems Center, San Diego. Work by CSC was performed under Contract N-00123-74-C-0312.

Released by
IP Lemaire, Head
Advanced Systems Division

Under authority of
HR Talkington, Head
Ocean Technology Department

18 NOSC

19 TR-146

UNCLASSIFIED

SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

REPORT DOCUMENTATION PAGE		READ INSTRUCTIONS BEFORE COMPLETING FORM
1. REPORT NUMBER	2. GOVT ACCESSION NO.	3. RECIPIENT'S CATALOG NUMBER
NOSC Technical Report 146 (TR 146)		
4. TITLE (and Subtitle)	5. TYPE OF REPORT AND PERIOD COVERED	
SIMULATION CODE FOR UNDERWATER TELEVISION SYSTEMS (SCOUTS)		1 Jun 76 - 1 May 77
6. AUTHOR(s)	7. SPONSORING AND GRANT NUMBER(s)	
Alan Gordon Thomas Selden	Naval Ocean Systems Center Computer Sciences Corporation	N 00123-74-C-0312
8. PERFORMING ORGANIZATION NAME AND ADDRESS	9. PROGRAM ELEMENT, PROJECT, TASK AREA & WORK UNIT NUMBERS	
Naval Ocean Systems Center San Diego, CA 92152	Program Element 63713N	
10. CONTROLLING OFFICE NAME AND ADDRESS	11. REPORT DATE	
Naval Sea Systems Command Washington, D.C.	11 Sep 77	
12. MONITORING AGENCY NAME & ADDRESS (if different from Controlling Office)	13. NUMBER OF PAGES	
	12 44b.	
14. DISTRIBUTION STATEMENT (of this Report)	15. SECURITY CLASS. (of this report)	
Approved for public release; distribution unlimited.		
16. DISTRIBUTION STATEMENT (of the abstract entered in Block 20, if different from Report)	17. SECURITY CLASS. (of this report)	
DD C JAN 19 1978 F		
18. SUPPLEMENTARY NOTES	19. KEY WORDS (Continue on reverse side if necessary and identify by block number)	
underwater television computerized simulation FORTRAN IV		
20. ABSTRACT (Continue on reverse side if necessary and identify by block number)	Based on the system performance equations developed in the Naval Undersea Center's NUC TP 303, Handbook of Underwater Imaging System Design, a computer code was written to predict the performance of underwater television systems. This code, known as SCOUTS (for "Simulation Code for Underwater Television Systems") is written in FORTRAN IV and is suitable for demand or batch runs. SCOUTS allows the user to select environmental, geometric and hardware parameters. It can be used for system design as well	

DD FORM 1473 EDITION OF 1 NOV 65 IS OBSOLETE

UNCLASSIFIED

SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

390612

Dec

UNCLASSIFIED

SECURITY CLASSIFICATION OF THIS PAGE(When Data Entered)

20. Continued.

as analysis. SCOUTS includes a parametric mode which allows automatic variation of multiple parameters. A complete source listing is included as an Appendix.

UNCLASSIFIED

SECURITY CLASSIFICATION OF THIS PAGE(When Data Entered)

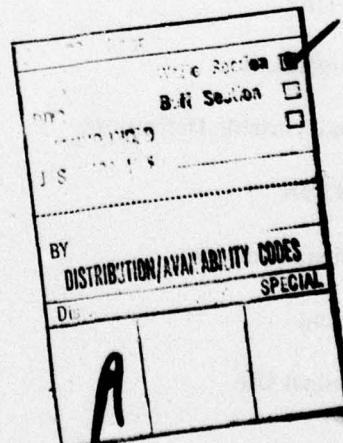
SUMMARY

PROBLEM

Using the system performance equations developed in NUC TP 303, Handbook of Underwater Imaging System Design, develop a computer code for predicting the performance of underwater television systems.

RESULTS

A code called SCOUTS (for "Simulation Code for Underwater Television Systems") and suitable for batch or demand runs has been written in FORTRAN IV. SCOUTS allows the user to select environmental, geometric, and hardware parameters. It can be used for system design as well as analysis. SCOUTS includes a parametric mode that allows automatic variation of multiple parameters.



CONTENTS

1.0 INTRODUCTION	5
1.1 Need for Performance Modeling of Underwater Television	5
1.2 The Handbook of Underwater Imaging System Design	5
1.3 Evolution of SCOUTS	6
2.0 OVERVIEW OF SCOUTS	7
2.1 System Geometry	7
2.2 User Inputs	8
2.3 Types of Calculations Available	10
2.4 Output	12
3.0 BASIS OF SCOUTS' CALCULATIONS	15
3.1 Water Characteristics	15
3.2 Hardware Characteristics	15
3.3 Performance Calculations	16
4.0 PROGRAM STRUCTURE	17
4.1 Subroutine Organization	17
4.2 Subroutine and Variable Definitions	18
5.0 PROGRAM OPERATION	21
5.1 Preliminary Requirements	21
5.2 Batch Processing	22
5.3 Demand Terminal Use	23
REFERENCES	24
APPENDIX	25

LIST OF FIGURES

1. SCOUTS system geometry.	7
2. Sample input for demand run.	9
3. Intermediate summary.	13
4. Final summary.	14
5. Program hierarchy.	17
6. Creation of absolute element.	21
7. Sample deck setup for batch execution of VSEARCH.	23

LIST OF TABLES

1. User inputs.	8
2. Variable names and definitions.	19
3. SCOUTS subroutines.	20
4. Batch mnemonics.	22

1.0 INTRODUCTION

This report documents the Naval Ocean Systems Center's (NOSC) Simulation Code for Underwater Television Systems (SCOUTS). SCOUTS is a real time, interactive (when used in the DEMAND mode) computer code which is useful in the design, analysis and performance prediction of a large class of underwater television systems. A complete printout of SCOUTS in the FORTRAN IV language is included in the Appendix. For those with immediate applications, the information in Sections 2 and 5 is sufficient to describe the operation and use of SCOUTS. Sections 3 and 4 give the theoretical basis for the calculations and internal program structure, respectively. The remainder of this Section concerns itself with the background and rationale of SCOUTS' development.

1.1 NEED FOR PERFORMANCE MODELING OF UNDERWATER TELEVISION

During the past decade and a half man has rapidly extended his presence beneath the surface of the ocean. Manned submersibles, tethered vehicles, moored drill rigs and even underwater habitats reflect his desire to understand and utilize the unique world beneath the ocean's surface. Underwater photography has been used throughout this era to document subsurface missions. This time period has also seen the development of underwater television to the point that it is equal in importance to photography as an undersea visual sensor.

Modern underwater television systems use lenses, light sources and cameras specifically designed for this environment. These components are incorporated into system concepts which attempt to minimize backscatter while maintaining an acceptable signal-to-noise ratio. As a result of the progress made during the last 15 years, the designer of underwater television systems now can choose from a wide variety of components and techniques in tailoring the system to his particular requirements.

One rather disturbing effect of the availability of advanced components and techniques was that predicting the performance of the resulting system became much more difficult. Simple single scattering models gave erroneous results at the longer ranges corresponding to advanced imaging systems. More exact models were available, but they were primarily theoretical and not generally suitable for engineering use. Thus, by the early 1970s, a situation arose where hardware was available which appeared to promise improved system performance, but the methodology for comparing the available hardware and accurately predicting performance was lacking.

1.2 THE HANDBOOK OF UNDERWATER IMAGING SYSTEM DESIGN

Under the sponsorship of the Navy's Deep Ocean Technology Program, the Naval Ocean Systems Center undertook the task of providing a definitive handbook for the designer of underwater television systems. This effort produced the Handbook of Underwater Imaging System Design (HUISD) (Reference 1). HUISD provided a detailed discussion of optical water parameters, propagation of light underwater, characteristics of system components and

advantages and drawbacks of system concepts then in use or proposed. More importantly, HUISD gave a detailed design procedure for obtaining measures of system performance based on the above factors. This design procedure, validated by comparative system tests performed by NOSC at Morris Dam in 1970, was presented both as a set of nomograms and a step-by-step series of equations. These equations, which were more accurate and flexible than the nomograms, required lengthy hand calculations and were tedious to implement. Consequently, these system equations were not applied as widely as had been hoped, though their predictions proved accurate in those instances where the hand calculations were performed.

1.3 EVOLUTION OF SCOUTS

About a year after the publication of HUISD, NOSC began work on the computer modeling portions of the Advanced Unmanned Search System (AUSS) program. Part of the AUSS model required the simulation (i.e., performance prediction) of underwater television systems. Since the HUISD equations were already developed and validated, these equations were used as the basis for the AUSS computer code. Only the HUISD performance analysis of conventional systems (i.e., systems using non-laser sources and raster-scanned receivers) was coded, since these systems were the ones most commonly in use. The first AUSS TV program was coded in BASIC, but in order to achieve faster operating times the program was rewritten in the FORTRAN language.

Experience with the AUSS TV program quickly showed that the HUISD system performance equations, coupled with real time, interactive demand terminal computing, provided a powerful tool for TV system design and analysis. Using the program the operator was immediately furnished with the performance of the selected configuration as well as indications as to what (e.g., source power, backscatter) were the limiting factors. By successively adjusting the TV system's configuration the operator could optimize it for his particular arrangement. This procedure was successfully used in designing and validating systems used in search, small-object detection, obstacle avoidance, etc.

One of the problems with the AUSS TV program was that it is a subprogram integral to the much larger AUSS model. In order to use just the TV portion, much information extraneous to the operation of the TV subsystem has to be keyed in. Because there was a significant number of AUSS users who were interested primarily in the TV subprogram, it was decided to develop SCOUTS as an independent computer code for the simulation of underwater television.

SCOUTS differs in a few important ways from the AUSS TV subprogram. SCOUTS is coded in FORTRAN IV and should be compatible as written with most medium- and large-size computers. A parametric capability allows several variables to be automatically varied according to limits defined by the user. Finally, although SCOUTS is primarily demand terminal-oriented, a batch option is included for computer facilities lacking real time capabilities.

2.0 OVERVIEW OF SCOUTS

This section will be concerned with defining the problem which SCOUTS solves, the inputs required and the outputs provided. Coupled with the detailed operating instructions in Section 5, this section provides sufficient information for successful operation of the program.

2.1 SYSTEM GEOMETRY

The geometry assumed by SCOUTS for its calculations is shown in Figure 1. The source and receiver are spatially separated by the source-receiver separation, d . Without loss of generality the source is assumed to be to the receiver's right when looking out towards the target. Their optical axes are coplanar and each is "canted in" by an angle δ so that a perpendicular dropped from the axes' intersection point will bisect the line joining the source and receiver. The receiver's (i.e., TV's) horizontal scan direction is parallel to the line joining source and receiver.

The target's center is in the plane containing the source and receiver's optical axes (i.e., the plane of the paper). The long dimension of the object is in the plane which is orthogonal to the plane of the paper and parallel to the line joining source and receiver. The

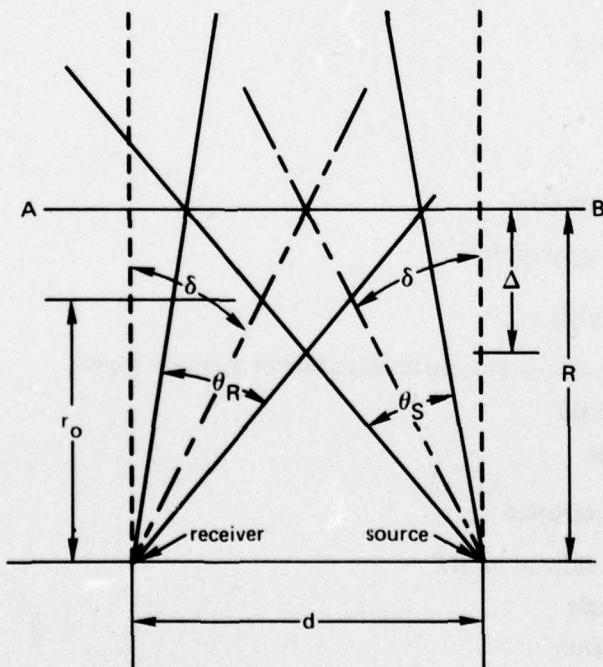


Figure 1. SCOUTS system geometry.

point of the object most distant from the source-receiver plane (i.e. the plane containing the source and receiver and perpendicular to the plane containing their optical axes) is assumed to be situated a distance R from it. The height of the object, H , is the extent of the object perpendicular to the source-receiver plane.

The distance r_0 in Figure 1 is the distance past which backscatter contributes to the image. The depth of field, Δ , is the maximum distance from the target plane at which there is a "common volume" where source and receiver beams intersect any portion of the target whose height exceeds Δ will not be imaged.

2.2 USER INPUTS

Table 1 contains the inputs necessary to define the imaging problem for SCOUTS. All these variables must be entered on the initial calculation of any run. A rewrite capability (see Section 2.5) is included so that this information does not have to be entered each time it is desired to change only some of the variables. An example of the initial input for a demand run is shown in Figure 2.

Table 1. User inputs.

Environment
1. Water type: coastal or deep
2. Peak-to-trough wave height
3. Peak-to-trough bottom roughness
Target Dimensions
4. Target length
5. Target height
Platform Characteristics
6. Source-receiver separation
Source Characteristics
7. Source type: incandescent, thallium iodide or mercury vapor
8. Source input power
9. Source full angle
Receiver Characteristics
10. Receiver type: vidicon or SIT
11. Receiver full angle
12. Optics transmission
13. F-number

CONVENTIONAL TV SYSTEM

ENVIRONMENT
COASTAL OR DEEP, 1 OR 2:?
>1
WAVE HEIGHT-PEAK TO TROUGH, FT:?
>2.
BOTTOM ROUGHNESS-PEAK TO TROUGH, FT:?
>.5
TARGET DIMENSIONS
LENGTH OF TARGET, FT:?
>10.
HEIGHT OF TARGET, FT:?
>.5
PLATFORM CHARACTERISTICS
SOURCE RECEIVER SEPARATION, FT:?
>1.
SOURCE CHARACTERISTICS
INCANDESCENT, THALLIUM IODIDE, OR MERCURY VAPOR; 1, 2 OR 3:?
>2
SOURCE POWER, WATTS:?
>200.
FULL SOURCE ANGLE, DEG:?
>40.
RECEIVER CHARACTERISTICS
VIDICON OR SIT, 1 OR 2:?
>1
FULL RECEIVER ANGLE, DEG:?
>40.
TRANSMISSION OF OPTICS:?
>.9
F-NUMBER:?
>1.5

Figure 2. Sample input for demand run.

Table 2 is for the most part self-explanatory; however, a few comments are in order. SCOUTS was originally written for an object lying on the bottom viewed from a TV camera which was responding to ocean heave. To ensure an adequate depth of field, Δ , SCOUTS chooses Δ so that

$$\Delta = H + 0.5 * (h_1 + h_2) \quad (1)$$

where h_1 and h_2 are the wave height and bottom roughness respectively. For situations where the target is not on the bottom, h_1 and h_2 should be set to zero.

SCOUTS incorporates information from HUISD to provide the appropriate characteristics for the water, lamp and receiver types. These characteristics will be discussed in detail in Section 3. However, in order for the user to supply meaningful inputs, it is necessary to discuss some of these characteristics here.

The two water types, coastal and deep, have optical characteristics identical with those plotted in Figures 6.2(B) and 6.2(C) of HUISD. The coastal water has a minimum attenuation, α , of $0.253/m$ ($1/\alpha = 3.95$ m) at $\lambda = 540$ nm. For the deep water, the corresponding numbers are $\alpha = 0.049/m$ ($1/\alpha = 20.4$ m) at 475 nm. The scattering coefficient, s , is $0.238/m$ for the coastal water and $0.030/m$ for deep water.

For sources with conical beam patterns, the required "full source angle," θ_s , is just the apex angle of the in-water beam pattern. For a nonconical source, the "equivalent conical beam pattern" (HUISD, p. 6-11, Eq. 6.10) must be determined and its apex angle used as input. For "source power," SCOUTS requires the electrical power input to the lamp, then computes the light power output.

The full receiver angle, θ_R , is the full in-water horizontal field of view of the TV camera. SCOUTS assumes the standard 4-by-3 aspect ratio for the receiver's field of view.

For transmission of optics SCOUTS requires the overall decimal efficiency of the source and receiver. This is the combined efficiency of all the optical elements following the source and up to and including the camera lens. For example, if the collection efficiency of the light source and the transmissivity of the camera lens were each 0.9, the required optical transmissivity would be 0.81.

The f-number, $f/$, is defined for small angles by

$$f/ = \frac{f}{D} \quad (2)$$

where f is the focal length and D the receiver aperture diameter. The $f/$ has a minimum value of 0.5. Most camera lenses will indicate the $f/$ corresponding to each aperture stop.

2.3 TYPES OF CALCULATIONS AVAILABLE

2.3.1 Demand and Batch Runs

SCOUTS can be run either as a batch or demand run. It is important to remember that SCOUTS was designed for real time, interactive computing, i.e. for demand runs. Design problems are most efficiently handled in the demand mode, since the user will be able to refine his design based on immediate feedback of the performance of his previous choice. Another advantage is that after a period of running design problems via demand runs, the user will develop his intuition for good underwater systems design.

The batch run capability was included primarily for computer systems not having real time facilities. However, the batch mode can be used to advantage in some situations. Some computer facilities charge substantially less for batch runs than for real time operation. Although SCOUTS' costs are low because less than 1.5 sec combined CAU (central arithmetic unit) and ER (executive request) time per calculation is required on the NOSC Univac 1110, the cost savings might be significant if many calculations are required. Batch runs necessarily provide a hard-copy output; this output may be absent or of poor quality on demand terminals. In very large runs using the parametric mode (see Section 2.3.2) where many variables are to be parametrized in a noninteractive fashion, using the batch mode allows the program to run without the operator being present.

2.3.2 Nonparametric and Parametric Modes

Nonparametric and parametric modes are available for both demand and batch runs. In the nonparametric mode, each input variable (see Table 1) takes on only one value and a calculation is immediately performed. In the parametric mode, the user selects one or more variables which he parametrizes by assigning each a beginning value, incremental step and final value. SCOUTS then performs as many calculations as necessary to evaluate the system for all the selected values of the input variables, and the results are output.

The parametric mode is thus similar to a succession of nonparametric calculations. The user selects the mode appropriate for his objectives. In optimizing a system, a succession of nonparametric calculations will allow the user to select new input values based on the results of past calculations. The performance of a selected system for various values of input data can be documented more quickly in the parametric mode.

2.3.3 Calculational Modes

There are three calculational modes available to the user: single range, maximum range or maximum swath width. The calculational mode is chosen in demand runs by the appropriate response to a SCOUTS question and in batch runs by inclusion of the appropriate mnemonic (see Table 4). In the single-range mode the system performance is evaluated at the range supplied by the user. To understand the other two modes we must first define what is meant by usable swath width.

The usable (or actual) swath width at range R is measured along that line in the source-receiver-target plane which is parallel to the line joining source and receiver and a distance R from it (the line AB in Figure 1). A segment of this line will have the following two properties: (1) a target located on this segment will have an image contrast greater than or equal to 0.07 and (2) the target's image will be large enough to span eight or more resolved TV lines. The length of this segment is taken to be the actual swath width. The actual swath width thus represents the lateral dimension in which successful imagery is possible.

As a function of range the swath width increases almost linearly with range, reaches a maximum and then drops off rapidly as the range is increased further. The maximum range calculation evaluates the system at that range where the swath width, after having passed its maximum, drops to zero. Similarly, the maximum swath width calculation evaluates the system at the range which yields the greatest swath width.

In the nonparametric mode, SCOUTS can evaluate the system in any of the three calculational modes. In the parametric mode, the user can have the system evaluated at either the maximum range or maximum swath width by indicating his choice and setting the range equal to zero. If the range has some other value the system will be evaluated at the selected range; however, SCOUTS will, at the user's option, calculate the value of the maximum range, maximum swath width, or both. Since SCOUTS performs these maximizing calculations by performing a series of single-range calculations, running time and costs can be minimized by selecting maximizing calculations only when necessary.

2.4 OUTPUT

2.4.1 Minimum Range

If the target is located too close to the source-receiver plane TV performance can be degraded. At distances which are too close, the resultant depth of field becomes less than the minimum value required by Eq. 1. Additionally, at distances sufficiently short the source will be pointing directly towards the receiver and blind it. SCOUTS therefore calculates a minimum range prior to evaluating the user's system. In demand runs this information is presented to the user in time for the user to choose a range greater than the minimum. SCOUTS then checks that the input range is indeed greater than the minimum. If not, the calculation will not be performed.

One of the disadvantages of batch runs is that the user does not receive an indication of the allowable minimum range until the run is over. Thus, there is an uncertainty as to the smallest range that will execute successfully. Since SCOUTS sets the minimum range no smaller than the source-receiver separation, batch input ranges should be larger than this value. This will not guarantee a successful run but is the best the user can do with no other *a priori* information.

2.4.2 Summary Output

The results of a SCOUTS calculation (or calculations in the parametric mode) represent the performance evaluation of the specified TV system. This performance evaluation is provided by SCOUTS in the form of two output summaries, the intermediate summary (Figure 3) and the final summary (Figure 4). Intermediate summaries are provided at the end of each demand nonparametric run, while final summaries are output in all modes. Both summaries contain essentially the same information in different formats. The meaning of each of the outputs will now be presented in the order in which they appear on the final summary.

The first thirteen columnar entries on Figure 3, from WATER TYPE to F-NUMBER, reproduce the user-selected input so that a record of the evaluated system is available along with the output. RANGE, FT and RANGE, AL give the range at which the calculation was performed in feet and attenuation lengths, respectively. All the performance measures from AVAILABLE LINES AT CTR to the second to the last entry refer to these measures evaluated at this range. MINIMUM, FT is the minimum allowable range (See Section 2.4.1) in feet. If the maximum range or maximum swath width options are chosen, the range at which each of these occurs is printed out in feet after MAXIMUM, FT and BEST, FT, respectively.

The number of resolved TV lines spanning the object length is given by AVAILABLE LINES AT CTR for the object located at the center of the receiver's field of view and by AT EDGE for the object located at the right-hand edge of the receiver's field of view. Because of the interpolation subroutine used, the number of lines may be in excess of 525, which is taken to be the display limit. It has been shown that eight lines represent the threshold of object recognition (Reference 3).

SCOUTS calculates the image contrast for an object having an inherent contrast of 50 percent and a highlight reflectivity of 75 percent. CONTRAST AT CENTER and AT EDGE are respectively the image contrast for a target located at the center and right-hand edge of

CONVENTIONAL TV SYSTEM

WATER TYPE:	COASTAL	SOURCE:	THAL. IOD.
WAVE HEIGHT, FT:	2.00000	SOURCE POWER:	200.00000
BOTTOM, FT:	.50000	SOURCE BEAM:	40.00000
LENGTH, FT:	10.00000	RECEIVER:	VIDICON
HEIGHT, FT:	.50000	RECEIVER BEAM:	40.00000
TAU:	.90000		
S.R. SEP.:	1.00000	F/:	1.50000
RANGE, FT:	24.40228	RANGE, AL:	1.87424
AVAILABLE LINES AT CTR.:	393.90030	AT EDGE:	393.90030
CONTRAST AT CENTER:	.16397	AT EDGE:	.07523
AVAILABLE L H S SW WIDTH =	8.81963 FT	ACTUAL =	8.22299 FT
LIMITED BY:R. ANG			
AVAILABLE R H S SW WIDTH =	8.95223 FT	ACTUAL =	8.22299 FT
LIMITED BY:S. ANG			
AVAILABLE SW WIDTH =	17.77186 FT	ACTUAL =	16.44598 FT

Figure 3. Intermediate summary.

the receiver's field of view. Since backscatter is most severe towards the right edge of the field of view (remember that SCOUTS assumes the light is to the right of the receiver) the contrast at this point (as well as the number of lines available) will always be less than that in the center.

The next eight entries on the final summary give information on the swath width to be expected. AV L.H.S. (R.H.S.) SW WIDTH, FT is the available left-hand (right-hand) swath width. This is the segment, in feet, of the swath width line (See Section 2.3.3) to the left (right) of the center of the receiver's field of view which is illuminated by both source and receiver beams. ACTUAL, FT is the portion of the available left-hand (right-hand) swath width, in feet, where the target may be successfully imaged. Again, the three criteria used by SCOUTS for successful imaging are: (1) adequate depth of field as defined by Eq. (1), (2) at least eight resolved TV lines across the target's longest dimension and (3) an image contrast of at least 7 percent.

LIMITED BY gives a mnemonic which indicates what is limiting the extent of the R.H.S. (L.H.S.) actual swath width. The mnemonics S. ANG. and R. ANG refer to source angle and receiver angle respectively and indicate that the corresponding beam's geometry is limiting the available swath width. BKSCTR indicates that the available swath exceeding 7-percent contrast is the limiting factor. Similarly, PWR signifies that the available swath width was limited by the requirement for eight TV lines/object length. DISPLAY is printed when the number of lines/raster height required to place eight TV lines across the image's longest dimension exceeds 525.

AV SWATH WIDTH, FT is the arithmetic sum of the available left- and right-hand swath width and indicates the total available swath width for the given geometry and range. ACTUAL, FT is the corresponding sum of the actual swath widths, which is the linear

WATER TYPE:	COASTAL	COASTAL	COASTAL	COASTAL
WAVE HEIGHT, FT:	2.0000	2.0000	2.0000	2.0000
BOTTOM ROUGHNESS, FT:	.5000	.5000	.5000	.5000
TARGET LENGTH, FT:	10.0000	10.0000	10.0000	10.0000
TARGET HEIGHT, FT:	.5000	.5000	.5000	.5000
S.R.SEP., FT:	1.0000	1.0000	2.0000	3.0000
SOURCE:	THAL. IOD.	THAL. IOD.	THAL. IOD.	THAL. IOD.
SOURCE POWER, WATTS:	200.0000	200.0000	200.0000	200.0000
SOURCE BEAM, DEG:	40.0000	40.0000	40.0000	40.0000
RECEIVER:	VIDICON	VIDICON	VIDICON	VIDICON
RECEIVER BEAM, DEG:	40.0000	40.0000	40.0000	40.0000
TAU:	.9000	.9000	.9000	.9000
F-NUMBER:	1.5000	1.5000	1.5000	1.5000
RANGE, FT:	24.4023	24.4023	29.7573	33.2921
RANGE, AL:	1.8742	1.8742	2.2855	2.5570
MINIMUM, FT:	2.5837	2.5837	3.0139	3.2687
MAXIMUM, FT:	.0000	.0000	.0000	.0000
BEST, FT:	24.4023	24.4023	29.7573	33.2921
AV LINES AT CTR:	393.9003	393.9003	322.7560	288.1938
AT EDGE:	393.9003	393.9003	322.7560	271.0501
CONTRAST AT CTR:	.1640	.1640	.1536	.1469
AT EDGE:	.0752	.0752	.0659	.0601
AV L.H.S. SW WIDTH, FT:	8.8196	8.8196	10.7119	11.9459
ACTUAL, FT:	8.2230	8.2230	10.1408	11.3969
- LIMITED BY:	R. ANG	R. ANG	R. ANG	R. ANG
AV R.H.S. SW WIDTH, FT:	8.9522	8.9522	10.9773	12.3444
ACTUAL, FT:	8.2230	8.2230	10.1408	10.2479
- LIMITED BY:	S. ANG	S. ANG	S. ANG	BKSCTR
AV SWATH WIDTH, FT:	17.7719	17.7719	21.6893	24.2904
ACTUAL, FT:	16.4460	16.4460	20.2816	21.6448
BEST, FT:	16.4460	16.4460	20.2816	21.6448

Figure 4. Final summary.

measure of the total horizontal distance over which satisfactory imagery is possible. **BEST**, FT is the maximum actual swath width. Zero will appear for this quantity unless the maximum swath width option has been selected.

3.0 BASIS OF SCOUTS CALCULATIONS

This section will outline the main features of the SCOUTS performance analysis and relate them to the corresponding steps in HUISID. This information is not strictly necessary for the general SCOUTS user, but it is useful to those who might desire to alter some of the program's internal parameters to suit their specific problems. To that end, the various water and hardware characteristics used by SCOUTS will be reviewed in Sections 3.1 and 3.2. Section 3.3 will provide an outline of the calculational flow keyed to the labeled line numbers of the detailed listing in the Appendix.

3.1 WATER CHARACTERISTICS

All the water characteristics used by SCOUTS are documented and referenced in HUISID and appear as stored matrices entered via DATA statements in the program MAIN. The following discussion of the program's features requires reference to the appendix, where the subroutines are arranged in alphabetical order. MAIN is the main program; the rest are subroutines. The function of each is explained in Section 4.

The array A(I, J) (line MAIN 21) stores the attenuation coefficient for coastal (I=1) and deep (I=2) water for forward (J=1) and backscattered (J=2) light. It was obtained from HUISD (Table 6.12). The array B(I,J,K) (MAIN 24) contains the ratio of the effective to actual attenuation coefficient indexed according to water type (I), forward or reverse propagation (J) and full beam angle (K); these data also come from Table 6.12 in HUISD. The interpolation array of corresponding beam angles is stored in matrix T (MAIN 49). Scattering coefficients, which were also obtained from Table 6.12, are stored in matrix S (MAIN 46). For both coastal and deep waters a rear hemisphere scattering (i.e., backscattering) percentage of 2 percent is used (line TVFU 33).

3.2 HARDWARE CHARACTERISTICS

The spectral characteristics of the source, receiver and water enter into the HUISD systems performance analysis through the constructed functions G(λ) and H(λ), which refer to forward and backscattered light, respectively. These functions are stored in the SCOUTS arrays G(I,J,K) and H(I,J,K) where I, J, K refer to water, receiver and source types, respectively. These entires were computed by subtracting entries in Table 6.8 of HUISD in accordance with the effective bandwidths listed in Table 6.5.

The imaging capability of the two types of cameras are stored in the array FM (I, J) (MAIN 30), where I refers to receiver type (i.e. I = 1 = vidicon and I = 2 = SIT) and J indexes two constants per receiver type. For a given camera tube current, i, the number of resolved TV lines, N, is obtained according to

$$N = \frac{\ln(i/FM(I, 1))}{FM(I, 2)} . \quad (3)$$

The values of $FM(I, J)$ were obtained by fitting Eq. (3) to available camera tube data. For the SIT the data were taken from HUISD, Figure 6.4(E), and the vidicon data from Reference 2. In each case the 50-percent contrast curve was used.

3.3 PERFORMANCE CALCULATIONS

The first calculation SCOUTS performs for each system's performance analysis is the evaluation of the minimum range. This occurs in lines HLIM 25-47. HLIMIT sets the minimum range equal to the greatest of (1) the source-receiver separation, (2) the minimum range which gives the required depth of field and (3) the minimum range which precludes the source and receiver looking at each other.

All the other calculations are done in the subroutine TVFUNC and the routines called by it. After some initialization, TVFUNC sets the range equal to that desired for single-range calculation (line TVFU 53) or to 110 percent of the minimum range in the maximizing modes (line TVFU 55). Next, a check is performed to make sure the display limit is satisfied (TVFU 59-64). If not, the remaining calculations are skipped and a flag is set so that a warning that the display limit has been exceeded is printed in the summary.

In lines TVFU 72-120, SCOUTS evaluates the number of lines at the center and the edge of the field of view. If less than the minimum (i.e. eight) number of lines is obtained at the center of the field of view, SCOUTS assigns minimum left- and right-power-limited swath widths (TVFU 101-105). If there is no power limit even at the edge of the field of view, the maximum swath widths are associated with the power limit (TVFU 117-119). In the case where the eight-line limit is encountered at some angle other than the maximum or minimum, this angle is found in an iterative search (TVFU 87-99) and the appropriate swath widths are computed (TVFU 109-114). In every case, the photocathode current is obtained in subroutine DFUNC, which implements Eqs. 6.17 and 6.34 of HUISD. EFUNC evaluates the number of resolved TV lines, using the matrix $FM(I, J)$.

The effects of backscatter are accounted for in lines TVFU 121-127. Using the subroutine XFUNC, which evaluates the photocathode current due to backscattered light (Eq. 6.48 of HUISD), the contrast is calculated at the center and at the right-hand edge of the field of view. Using the interpolation routine, GFUNC, an appropriate right-hand swath width due to backscatter, $W(7)$, is obtained. The geometric limitations on the swath width are next obtained ($W(2)$, $W(3)$, $W(5)$, $W(6)$) through the use of UFUNC and VFUNC. FNK finds the limiting right and left swath widths and descriptors. The limiting swath widths are summed to yield the total actual swath width.

Lines TVFU 146-184 determine whether the calculation is for a single-range or a maximizing calculation. If it is for a single-range calculation, TVFUNC returns control to HLIMIT. If not, TVFUNC determines whether the calculation is complete. If the last range increment is less than 5 percent of an attenuation length, control is returned to HLIMIT. If not, the range is incremented or decremented depending on the previous value of the swath width, and the performance is recalculated starting at line TVFU 40.

4.0 PROGRAM STRUCTURE

4.1 SUBROUTINE ORGANIZATION

SCOUTS consists of a main program and seventeen subroutines. The main program defines constants used in performing the calculations and acts as a program driver. Eleven of the seventeen subroutines perform the computations. The remaining six subroutines – SUBROUTINE DATAIN, SUBROUTINE CONVT, SUBROUTINE HLIMIT, SUBROUTINE UPDATE, SUBROUTINE DATAOT, and SUBROUTINE SUMMARY – were developed to provide flexibility of input and output. Figure 5 illustrates program hierarchy. A printout of each subroutine and of the main program is provided in the appendix.

SUBROUTINE DATAIN reads user-supplied input data and sets program control flags. If batch processing is desired, the user-supplied data are input via the card reader (unit number 5). Beginning in column one, the first card of this data deck must contain the five-character alphanumeric word BATCH. This card sets the flag for batch processing. Subsequent data cards are then read in an A6, 3X, 3F10.0 format. The alphanumeric information is a mnemonic instruction that directs the program to a specific section of code. The data values, where applicable, represent the initial, incremental, and final values of the data variables. Default values are set prior to reading any user-supplied data.

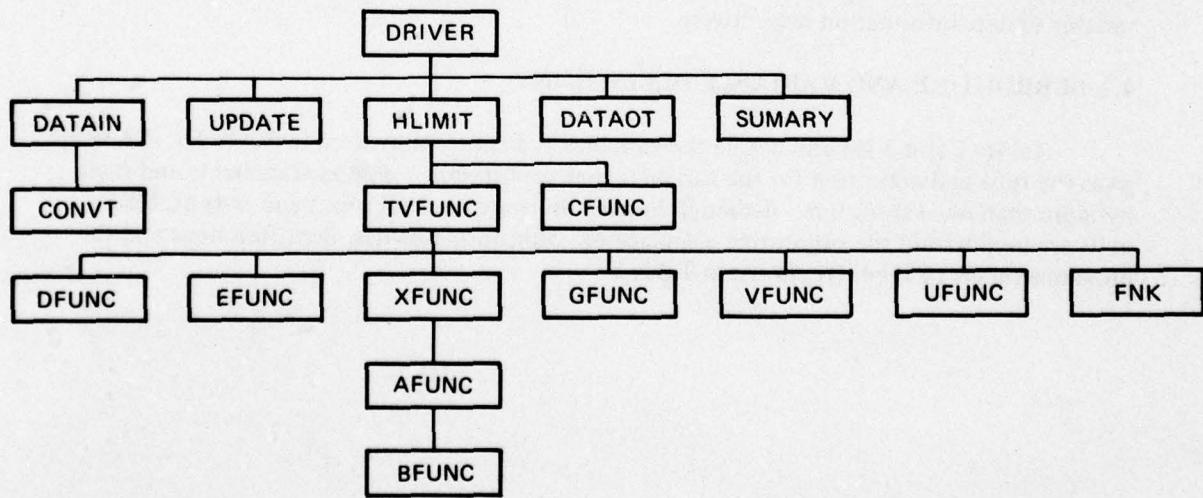


Figure 5. Program hierarchy.

If the program is being executed via a demand terminal, the terminal itself is the input device. In this case, interactive communication with the program is performed. The format of the input data, unless otherwise specified by the program, is G10.0. It should be noted that the program can be executed via a demand terminal without interactive communication by typing the word BATCH as the initial response. Data are input as though batch processing were being performed. (For more detailed information on input data, see the section under program operation.)

SUBROUTINE HLIMIT computes the minimum range and establishes the type of computations that will be made — single range, maximum range and/or maximum width.

Incrementing of the input data variables is performed by **SUBROUTINE UPDATE**. This subroutine is called only when parametric variables have been defined.

Two subroutines handle the printing of output data. Whenever execution is performed via a demand terminal, and no parametric variables have been defined, **SUBROUTINE DATAOT** prints the results after each evaluation is performed. This subroutine also writes a summary of the performance evaluation to the temporary disk file (unit number 9). At the conclusion of execution, **SUBROUTINE SUMMARY** reads the information stored on disk and prints tabular summaries of all performance evaluations computed. For batch processing, the printer (unit number 6) is the peripheral device utilized for output; for demand terminal processing, the terminal is the output device.

SUBROUTINE CONVT was developed to allow the demand terminal user to input integer values left-justified. If a two-digit integer value has been entered, the second digit is read as alphanumeric data and "converted" to the appropriate numeric value.

Data communication within the program is handled primarily through labeled common blocks. The four blocks established — OPTS, TVCOM, IOLIST, and UPD — store user-defined control options, program computing constants, input/output variables, and input variable update information respectively.

4.2 SUBROUTINE AND VARIABLE DEFINITIONS

Tables 2 and 3 list and define the variables and subroutines used in SCOUTS. Table 2 gives the type and definition for the variables that are listed in common statements and used by more than one subroutine. Although English units are used for input and output, MKS units are used within the subroutine calculations. Subroutine names, their functions and the programs called by them are shown in Table 3.

Table 2. Variable names and definitions.

Descriptive Variable	Type	Definition
NVIRON	integer	water type
WAVEHT	real	wave height
BTMRUF	real	bottom roughness
TARGLN	real	target length
TARGHT	real	target height
HEIGHT	real	height (or range)
SRSEP	real	source-receiver separation
SPOWR	real	source power
SDELF	real	full source angle
IRTPE	integer	receiver type
RDELF	real	full receiver angle
OPTRAN	real	optics transmission
FNUM	real	f-number
ISTPE	integer	source type
RANGMN	real	minimum range
RANGMX	real	maximum range
RANGEB	real	best range
SWWTHB	real	best swath width
HTUS	real	usual height
SWWTH	real	actual swath width
SWWTHL	real	actual L.H.S. swath width
SWWTHR	real	actual R.H.S. swath width
LIMFLG	integer	display limit flag
NDEXWL	integer	L.H.S. width index
NDEXWR	integer	R.H.S. width index
LINCEN	real	available lines at center
LINEDG	real	available lines at edge
CTRCEN	real	contrast at center
CTREDG	real	contrast at edge

Table 3. SCOUTS subroutines.

Subprogram	Function	Reference	Referenced by
MAIN	program driver; defines computing constants	DATAIN, HLIMIT, DATAOT, UPDATE, SUMARY	—
DATAIN	reads input data; sets control flags	CONVT	MAIN
HLIMIT	computes minimum range, defines type of computation	CFUNC, TVFUNC ATAN, TAN	MAIN
DATAOT	prints results after each evaluation; writes summary data to disk	—	MAIN
UPDATE	increments parametric input variables	—	MAIN
SUMARY	prints tabular summaries	—	MAIN
TVFUNC	main computational subroutines	DFUNC, EFUNC, VFUNC, XFUNC, GFUNC, UFUNC, FNK COS, ATAN, TAN	HLIMIT
CONVT	converts a single character to numeric data	—	DATAIN
XFUNC	calculates backscatter current	AFUNC	TVFUNC
AFUNC	Simpson's rule integration routine for backscatter integral	BFUNC, EXP, ALOG	XFUNC
BFUNC	integrand for backscatter integral	EXP	AFUNC
CFUNC	finds effective attenuation coefficient by interpolation	—	HLIMIT
DFUNC	computes signal current	TAN, ATAN, COS, EXP	TVFUNC
EFUNC	evaluates number of lines	ALOG	TVFUNC
FNK	interpolates for limiting width and description	—	TVFUNC
GFUNC	interpolates for backscatter width	TAN, ALOG	TVFUNC
UFUNC	calculates l.h.s. receiver and r.h.s. source widths	—	TVFUNC
VFUNC	calculates r.h.s. receiver and l.h.s. source widths	—	TVFUNC

5.0 PROGRAM OPERATION

This section is intended to provide the user with information that is required for successful execution of the visual search system program.

5.1 PRELIMINARY REQUIREMENTS

The visual search system program utilizes three peripheral devices for transmission of data: the card reader, the printer, and a disk. The card reader and the printer are assigned the standard FORTRAN unit numbers 5 and 6 respectively. Unit 9 is assigned to a disk file used temporarily to store unformatted data of summary output information. As required by the individual operating system involved, these unit numbers must be assigned to the appropriate peripheral device prior to compilation of the program.

After having assigned unit numbers to the appropriate peripheral device, the program is compiled and linked to create a set of absolute (or executable) binaries. Figure 6 illustrates the procedure used on the UNIVAC 1110 to create the absolute element for execution. In the example the absolutes are stored under the name VSEARCH of the file TVSENS. The program driver and each subprogram are stored under separate file element names.

```
@ FOR, N TVSENS. MAIN
@ FOR, N TVSENS. DATAIN
@ FOR, N TVSENS. CONVT
@ FOR, N TVSENS. HLIMIT
@ FOR, N TVSENS. TVFUNC
@ FOR, N TVSENS. UPDATE
@ FOR, N TVSENS. DATAOT
@ FOR, N TVSENS. SUMARY
@ FOR, N TVSENS. AFUNC
@ FOR, N TVSENS. BFUNC
@ FOR, N TVSENS. CFUNC
@ FOR, N TVSENS. DFUNC
@ FOR, N TVSENS. EFUNC
@ FOR, N TVSENS. FNK
@ FOR, N TVSENS. GFUNC
@ FOR, N TVSENS. UFUNC
@ FOR, N TVSENS. VFUNC
@ FOR, N TVSENS. XFUNC
@ PACK   TVSENS.
@ PREP   TVSENS.
@ MAP, N , TVSENS. VSEARCH
      IN   TVSENS.
      END
```

Figure 6. Creation of absolute element.

5.2 BATCH PROCESSING

Batch processing of the absolute element VSEARCH is performed whenever the user does not wish interactive communication with the program. If batch processing is desired, the first card image of input data must be the Hollerith string BATCH. This five-character string informs the program that subsequent input will be in an A6, 3X, 3F10.0 format. The alphanumeric information is a mnemonic instruction that directs the program to a specific section of code. The data values, where applicable, represent the initial, the incremental, and the final values of the appropriate input variable. Table 4 lists the mnemonic instructions, the variable defined, and their default values. It should be noted that although the data assignment instructions may occur in any order within the data deck, the STOP instruction must follow any given set of data assignment instructions. The last card image of the data deck must be the mnemonic instruction FINISH. Figure 7 illustrates a sample deck setup for execution of VSEARCH through batch processing.

Table 4. Batch mnemonics.

Mnemonic	Defines	Default Value
NVIRON	=1., coastal; =2., deep	1.
ISTPE	=1., incandescent; =2., thallium; =3. mercury vapor	1.
IRTPE	=1., vidicon; =2., SIT	1.
WAVEHT	wave height	0.
BTMRUF	bottom roughness	0.
TARGLN	target length	0.
TARGHT	target height	0.
SRSEP	source-receiver separation	0.
SPOWR	source power	0.
SDELF	full source angle	0.
RDELF	full receiver angle	0.
OPTRAN	optic transmission	0.
FNUM	f-number	.5
HEIGHT	range (or height)	0.
MAXSW	sets flag to compute maximum width	—
MAXRAN	sets flag to compute maximum range	—
SINGRN	sets flag to compute single range	—
STOP	stop reading input; end-of-record indicator for input	—
FINISH	stop execution; end-of-file indicator for input	—

BATCH			
NVIRON	1.00000	.00000	.00000
WAVEHT	2.00000	.00000	.00000
BTMRUF	.50000	.00000	.00000
TARGLN	10.00000	.00000	.00000
TARGHT	.50000	.00000	.00000
SRSEP	1.00000	.00000	.00000
ISTPE	2.00000	.00000	.00000
SPOWR	200.00000	.00000	.00000
SDELF	40.00000	.00000	.00000
IRTPE	1.00000	.00000	.00000
RDELF	40.00000	.00000	.00000
OPTRAN	.90000	.00000	.00000
FNUM	1.50000	.00000	.00000
MAXSW	.00000	.00000	.00000
STOP	.00000	.00000	.00000
NVIRON	2.00000	.00000	.00000
RANGE	65.00000	10.00000	105.00000
MAXSW	.00000	.00000	.00000
STOP	.00000	.00000	.00000
SRSEP	1.00000	1.00000	4.00000
RANGE	.00000	.00000	.00000
MAXSW	.00000	.00000	.00000
STOP	.00000	.00000	.00000
SRSEP	5.00000	.00000	.00000
SPOWR	50.00000	50.00000	105.00000
MAXSW	.00000	.00000	.00000
STOP	.00000	.00000	.00000
SDELF	20.00000	.00000	.00000
RDELF	20.00000	.00000	.00000
MAXSW	.00000	.00000	.00000
STOP	.00000	.00000	.00000
FINISH	.00000	.00000	.00000

Figure 7. Sample deck setup for batch execution of VSEARCH.

5.3 DEMAND TERMINAL USE

SCOUTS should be executed via a demand terminal whenever the user wishes interactive communication. A demand run also provides intermediate information that can be useful in determining which input variables should be changed.

Execution of the absolute element is performed by responding to the program's questions. Questions concerning environment, source type and receiver type are answered in an I1 format. Other initial input is in a G10.0 format. After the initial input, SCOUTS asks for direction as to parametric or nonparametric, the calculational mode, whether another run is desired, etc. These questions are also answered in an I1 format.

It should be noted that batch processing of SCOUTS can be performed from a demand terminal by responding 'BATCH' to the first question asked. In this case the succeeding input will have to follow exactly the format of Figure 7.

REFERENCES

1. Naval Undersea Center. NUC TP 303, Handbook of Underwater Imaging System Design, by C. J. Funk, S. B. Bryant and P. J. Heckman, Jr. July 1972.
2. Biberman, L. M. and S. Nudelman. Photoelectronic Imaging Devices, Vol. 2, p. 539. Plenum Press, New York, N. Y., 1971.
3. RCA Commercial Engineering. RCA Electro-Optics Handbook, p. 121. Harrison, N. J., 1974.

APPENDIX

This appendix contains a complete FORTRAN IV listing of SCOUTS, arranged in alphabetical order by subroutine name. MAIN is the main program; all the rest are FORTRAN subroutines. All programs have compiled successfully on the NOSC UNIVAC 1110.

```

000001 001      FUNCTION AFUNC(A)
000002 001      C
000003 001      C
000004 001      D3=-1.* ALOG(.00125*(EXP(-A)-EXP(-2.*A))/A)/A
000005 001      D1=n3
000006 001      10 U2=n3-ALOG(D1)/A
000007 001      IF (ARS((D2-D1)/U2).LT..01) GO TO 20
000008 001      D1=n2
000009 001      GO TO 10
000010 001      20 U0=(D2-1.)/2.
000011 001      D1=n0/3.
000012 001      D4=nFUNC(A+1.)*RFUNC(A+02)
000013 001      D7=nFUNC(A+1.*D0)
000014 001      U2=n1*(D4+4.*D7)
000015 001      D3=1.
000016 001      30 D3=2.*D3
000017 001      D1=n1/2.
000018 001      D0=n0/2.
000019 001      D9=nFUNC(A+1.*D0)
000020 001      D8=n9
000021 001      E1=EXP(-1.*A*2.*D0)
000022 001      J1=n3-1.
000023 001      DO 40 J=1,J1
000024 001      E2=(1.+(2.*FLOAT(J)-1.)*D0)/(1.+(2.*FLOAT(J)+1.)*D0)
000025 001      U8=n8*E1*E2*E2
000026 001      U9=n9*U8
000027 001      40 CONTINUE
000028 001      U5=n1*(D4+2.*D7+4.*D9)
000029 001      IF (ARS((D5-U2)/D5).LT..01) GO TO 50
000030 001      D2=n5
000031 001      D7=n9*D7
000032 001      GO TO 30
000033 001      50 AFUNC=U5
000034 001      RETURN
000035 001      ENQ

000001 001      FUNCTION BFUNC(A+B)
000002 001      BFUNC=(EXP(-A*B))/(B+B)
000003 001      RETURN
000004 001      ENQ

000001 001      FUNCTION CFUNC(TUIR,THET,IWAT)
000002 001      C
000003 001      C
000004 001      COMMON /TVCOM/ A(2,2), B(2,11,2), C(20), U(2,2), E(10),
000005 001      1 F(2,2), G(2,2,3), H(2,2,3), N(10), O(15), P(2,2),
000006 001      2 Q(15), S(2), T(11), W(10), Z(2), PII
000007 001      C
000008 001      DO 10 IT=1,11
000009 001      IF (THET.LT.T(11)) GO TO 20
000010 001      10 CONTINUE
000011 001      20 ID2=MIN(0,IT+1)
000012 001      II=ID2-1
000013 001      U5=R(IWAT+IT+IUIR)
000014 001      CFUNC=U5+((THET-T(11))/(T(ID2)-T(11))+(B(IWAT+ID2+IUIR)-R(IWAT+IUIR))
000015 001      1+IUIR)
000016 001      RETURN
000017 001      ENQ

000001 001      SUBROUTINE CONVT (IVAL)
000002 001      C
000003 001      C
000004 001      C
000005 001      C
000006 001      DIMENSION ICHAR(10)
000007 001      DATA ICHAR /1H1,1H2,1H3,1H4,1H5,1H6,1H7,1H8,1H9,1H0/
000008 001      C
000009 001      DO 10 I=1,10
000010 001      INUM=1
000011 001      IF (INUM.EQ.10) INUM=0
000012 001      IF (IVAL.EQ.ICHAR(I)) GO TO 20
000013 001      10 CONTINUE
000014 001      PRINT 40
000015 001      GO TO 30
000016 001      C
000017 001      20 IVAL=INUM
000018 001      30 RETURN
000019 001      C
000020 001      40 FORMAT (1$H NO VALUE FOUND)
000021 001      ENQ

```

000001	001			DAIN 1
000002	001	C		DAIN 2
000003	001	C		DAIN 3
000004	001	C		DAIN 4
000005	001	C		DAIN 5
000006	001	C		DAIN 6
000007	001	C		DAIN 7
000008	001	C		DAIN 8
000009	001	C		DAIN 9
000010	001	C		DAIN 10
000011	001	C		DAIN 11
000012	001	C		DAIN 12
000013	001	C		DAIN 13
000014	001	C		DAIN 14
000015	001	C		DAIN 15
000016	001	C		DAIN 16
000017	001	C		DAIN 17
000018	001	C		DAIN 18
000019	001	C		DAIN 19
000020	001	C		DAIN 20
000021	001	C		DAIN 21
000022	001	C		DAIN 22
000023	001	C		DAIN 23
000024	001	C		DAIN 24
000025	001	C		DAIN 25
000026	001	C		DAIN 26
000027	001	C		DAIN 27
000028	001	C		DAIN 28
000029	001	C		DAIN 29
000030	001	C		DAIN 30
000031	001	C		DAIN 31
000032	001	C		DAIN 32
000033	001	C		DAIN 33
000034	001	C		DAIN 34
000035	001	C		DAIN 35
000036	001	C		DAIN 36
000037	001	C		DAIN 37
000038	001	C		DAIN 38
000039	001	C		DAIN 39
000040	001	C		DAIN 40
000041	001	C		DAIN 41
000042	001	C		DAIN 42
000043	001	C		DAIN 43
000044	001	C		DAIN 44
000045	001	C		DAIN 45
000046	001	C		DAIN 46
000047	001	C		DAIN 47
000048	001	C		DAIN 48
000049	001	C		DAIN 49
000050	001	C		DAIN 50
000051	001	C		DAIN 51
000052	001	C		DAIN 52
000053	001	C		DAIN 53
000054	001	C		DAIN 54
000055	001	C		DAIN 55
000056	001	C		DAIN 56
000057	001	C		DAIN 57
000058	001	C		DAIN 58
000059	001	C		DAIN 59
000060	001	C		DAIN 60
000061	001	C		DAIN 61
000062	001	C		DAIN 62
000063	001	C		DAIN 63
000064	001	C		DAIN 64
000065	001	C		DAIN 65
000066	001	C		DAIN 66
000067	001	C		DAIN 67
000068	001	C		DAIN 68
000069	001	C		DAIN 69
000070	001	C		DAIN 70
000071	001	C		DAIN 71
000072	001	C		DAIN 72
000073	001	C		DAIN 73
000074	001	C		DAIN 74
000075	001	C		DAIN 75
000076	001	C		DAIN 76
000077	001	C		DAIN 77
000078	001	C		DAIN 78
000079	001	C		DAIN 79
000080	001	C		DAIN 80
000081	001	C		DAIN 81
000082	001	C		DAIN 82
000083	001	C		DAIN 83
000084	001	C		DAIN 84
000085	001	C		DAIN 85

SUBROUTINE DATAIN (ISTART,IEND)
 SUBROUTINE DATAIN READS INPUT DATA
 AND DEFINES VARIOUS COMPUTING CONSTANTS.
 COMMON BLOCK OF OPTIONS
 COMMON /UPI/ IRUN,IPARAM,ICAL
 COMMON BLOCK OF I/O VARIABLES
 COMMON /IOLIST/ NIVRON, ISIPE, IHTPE, WAVEHT, UTMRUF, TARGLN,
 1 TARGHT, SHSEP, SHWRF, SULF, RULF, UPTRAN, FNUM, HIGHT,
 2 RANGIN, RANGE, RANGAL, RANGR, SHWTHL, SHWTHR, SHWTHA,
 3 SHWTH, AVSOL, AVSWR, AVSW, LINEN, LINENL, CTRCEN, CTREDG,
 4 NUFXWL, NUFXHN, LIMFLG
 COMMON BLOCK OF COMPUTING CONSTANTS
 COMMON /IVCOM/ A(2,2), B(2,1,2), C(2,0), D(2,2), E(10),
 1 FM(2,2), G(2,2,3), H(2,2,3), N(10), O(15), P(2,2),
 2 Q(15), S(2), T(11), W(10), Z(2), PT1
 COMMON BLOCK OF UPDATE INFORMATION
 COMMON /UPI/ SIEP(14), FUVAL(14)
 DIMENSION ISIPE(19)
 REAL LINEN,LINENL
 DATA STATEMENT OF INSTRUCTIONS
 DATA ISINST/SHIVRON, SHISTPF, SHIKTPF, SHWAVLHT, SHATHRUF,
 1 SHARGLN, SHARTGNT, SHISHPF, SHSPDR, SHSULF, SHRULF,
 2 SHRULF, SHOPTRN, SHENUM, SHRANGE, SHMAXSH, SHMAXHN,
 3 SHMAXHN, SHISINGHN, SHSTOP, SHFINISH/
 DATA NIVRON,ISIPE,IHTPE,WAVEHT,UTMRUF,TARGLN,TARGHT,
 1 SHSEP,SHWRF,SULF,RULF,UPTRAN,FNUM /3+1900,0,5/
 DETERMINE THE TYPE OF PROCESSING, BATCH OR DEMAND. WHEN DAIN 37
 ISSTART, THE TYPE OF PROCESSING HAS ALREADY BEEN DETERMINED. DAIN 38
 IF (ISSTART,FG,0) GO TO 20
 IRUN=1
 PRINT 500
 FIRST CARD OF DATA DETERMINES THE FORMAT OF FOLLOWING
 INPUT - BATCH OR DEMAND TERMINAL.
 READ (5+510) RUN
 IF (RUN,LE,0,SHRATCH) GO TO 10
 RUN=SHUFMAN
 IRUN=0
 10 PRINT 520, RUN
 IF HATCH RUN, READ TO INPUT BY MNEMONIC INSTRUCTIONS. DAIN 51
 20 IF (IRUN,FG,1) GO TO 270
 DEMAND RUN
 IF PREVIOUS CALCULATIONS HAVE BEEN PERFORMED, RANCH
 TO CODE THAT DECIDES WHETHER OR NOT NEW CALCULATIONS ARE
 UPSTREAM. DAIN 56
 IF (ISSTART,FG,1) GO TO 210
 READ INITIAL VALUES OF INPUT DATA. IF ISSTART=1, PREVIOUS
 CALCULATIONS HAVE BEEN PERFORMED AND A CHANGE IS BEING MADE
 SPECIFIC INITIAL VALUES. DAIN 62
 PRINT 560
 30 PRINT 570
 READ (5+530) NIVRON
 IF (NIVRON,NE,1,AND,NIVRON,NE,2) GO TO 70
 IF (ISSTART,FG,1) GO TO 240
 40 PRINT 580
 READ (5+540) WAVEHT
 IF (WAVEHT,LT,0) GO TO 50
 IF (ISSTART,FG,1) GO TO 240
 50 PRINT 590
 READ (5+540) TARGLN
 IF (TARGLN,LT,0) GO TO 60
 IF (ISSTART,FG,1) GO TO 240
 60 PRINT 610
 READ (5+540) TARGHT
 IF (TARGHT,LT,0) GO TO 70
 IF (ISSTART,FG,1) GO TO 240
 70 PRINT 620
 READ (5+540) SHSEP
 IF (SHSEP,LT,0) GO TO 70
 IF (ISSTART,FG,1) GO TO 240

```

000006 001 PRINT 630
000007 001 BU PRINT 640
000008 001 READ (5+540) 5NSEP
000009 001 IF (ISPFPR.LT.0.) GO TO 80
000010 001 IF (ISTART.F0.1) GO TO 260
000011 001 PRINT 650
000012 001 BU PRINT 660
000013 001 READ (5+530) 1STPME
000014 001 IF (1SINE.LT.1.0N.1STPME.GT.3) GO TO 90
000015 001 IF (1STANT.F0.1) GO TO 260
000016 001 130 PRINT 670
000017 001 READ (5+540) 5NOWN
000018 001 IF (ISPFNL.LT.0.) GO TO 100
000019 001 IF (1STANT.F0.1) GO TO 260
000020 001 110 PRINT 680
000021 001 READ (5+540) 5NELF
000022 001 IF (1NELF.LT.0..0N.5NELF.GT.100.) GO TO 110
000023 001 IF (1STANT.F0.1) GO TO 260
000024 001 PRINT 690
000025 001 120 PRINT 700
000026 001 READ (5+530) 1NTPME
000027 001 IF (1NPL.NE.1.0N.1NTPME.NE.2) GO TO 120
000028 001 IF (1STANT.F0.1) GO TO 260
000029 001 130 PRINT 710
000030 001 READ (5+540) 5NUELF
000031 001 IF (1NUELF.LT.0..0N.5NUELF.GT.100.) GO TO 130
000032 001 IF (1STANT.F0.1) GO TO 260
000033 001 140 PRINT 720
000034 001 READ (5+540) 5NTRAN
000035 001 IF (1NTRAN.LT.0..0N.5NTRAN.GT.1.) GO TO 140
000036 001 IF (1STANT.F0.1) GO TO 260
000037 001 150 PRINT 730
000038 001 READ (5+540) 5NFM
000039 001 IF (1NFM.LT.5) GO TO 150
000040 001 IF (1STANT.F0.1) GO TO 260
000041 001 C
000042 001 C PARAMETRIC OR NON-PARAMETRIC MODE? IF NON-PARAMETRIC
000043 001 C MOUF (1PARAM.NE.1) VARIABLE INPUT COMPLETE.
000044 001 C PRINT 740
000045 001 C READ (5+530) 1PMAM
000046 001 C IF (1PARAM.NE.1) GO TO 260
000047 001 C
000048 001 C GIVE INDEX OF SPECIFIC PARAMETRIC VARIABLE. THEN READ
000049 001 C ITS INCREMENTAL AND ENDING VALUES.
000050 001 C PRINT 750
000051 001 C 160 READ (5+530) 1N1.1N2
000052 001 C IF (1N.F0.0) GO TO 160
000053 001 C IF (1N.G0.1N) GO TO 170
000054 001 C CAL1 COUNT (1N2)
000055 001 C 1N1.1N2.1N1.1N2
000056 001 C 170 PRINT 760
000057 001 C READ (5+540) 5NIPR(1N1)
000058 001 C PRINT 770
000059 001 C READ (5+540) 5NUVAL(1N1)
000060 001 C 180 IF (1STANT.F0.1) GO TO 260
000061 001 C
000062 001 C IF ADDITIONAL PARAMETRIC VARIABLES ARE DESIRED, CHANGE
000063 001 C
000064 001 C 1NUFA.
000065 001 C PRINT 780
000066 001 C READ (5+530) 1N2
000067 001 C IF (1N.G0.1) GO TO 190
000068 001 C PRINT 830
000069 001 C GO TO 1A0
000070 001 C
000071 001 C IF PARAMETRIC MODE, DETERMINE THE CALCULATIONS TO BE
000072 001 C PERFORMED.
000073 001 C READ THE INITIAL VALUE FOR COMPUTING A SINGLE RANGE, AND
000074 001 C THE CORRESPONDING INCREMENTAL AND ENDING VALUES.
000075 001 C 190 IF (1PARAM.NE.1) GO TO 840
000076 001 C PRINT 850
000077 001 C READ (5+530) 1CAL
000078 001 C IF (1CAL.GT.4) 1CAL=1
000079 001 C IF (1STANT.F0.1) GO TO 840
000080 001 C
000081 001 C VARIABLE INPUT COMPLETE.
000082 001 C 200 PRINT 7A0
000083 001 C 1STANT=1
000084 001 C GO TO 840
000085 001 C
000086 001 C PREVIOUS CALCULATIONS HAVE BEEN PERFORMED.
000087 001 C
000088 001 C IF ANOTHER RUN IS DESIRED, MAKE NECESSARY CHANGES (IF ANY) 210
000089 001 C READ (5+530) 1N2

```

```

000170 001 IF (IN2.NF.1) GO TO 490
000171 001 IF (IPARAM.FG.1) CALL SUMMARY (IRUN,ISUM)
000172 001 START=1
000173 001 PRINT 740
000174 001 READ (5+530) IPARAM
000175 001 PRINT 840
000176 001 READ (5+530) IN2
000177 001 IF (IN2.NF.1) GO TO 490
000178 001 C
000179 001 C GIVE INDEX OF VARIABLE TO BE CHANGED! THEN BRANCH TO
000180 001 C APPROPRIATE READ.
000181 001 PRINT 750
000182 001 250 READ (5+530) IN=IN2
000183 001 IF (IN=,ER,1H) GO TO 230
000184 001 CALI CONVT (IN2)
000185 001 IN=10*IN+IN2
000186 001 230 GO TO (30+60+120+40+50+60+70+80+100+110+130+140+150+240), IN
000187 001 C
000188 001 C IF PARAMETRIC MODE: BRANCH TO CODE THAT READS INCREMENTAL
000189 001 C AND ENDING VALUES.
000190 001 240 IF (IPARAM,ME.1) GO TO 260
000191 001 PRINT 820
000192 001 READ (5+530) IN2
000193 001 IF (IN2.NF.1) GO TO 250
000194 001 GO TO 170
000195 001 250 STEP(IN)=0.
000196 001 C
000197 001 C IF ADDITIONAL CHANGES ARE DESIRED, CHANGE INDEX.
000198 001 260 PRINT 810
000199 001 READ (5+530) IN2
000200 001 IF (IN2.NF.1) GO TO 190
000201 001 PRINT 830
000202 001 GO TO 220
000203 001 C
000204 001 C BATCH RUN.
000205 001 C
000206 001 C READ INPUT MNEMONIC. IF ANY INCREMENTAL VALUE APPEARS,
000207 001 C SET THE FLAG FOR PARAMETRIC INPUT.
000208 001 C
000209 001 270 READ (5+510) IVAR+21+72+23
000210 001 WRITE (6+550) IVAR+71+22+23
000211 001 C
000212 001 IF (Z1.LT.0.) Z1=0.
000213 001 IF (ARS(27).ER.0.) Z3=21
000214 001 IF (ARS(27).NF.0.) IPARAM=1
000215 001 C
000216 001 C TEST THE MNEMONIC AGAINST THE SET OF INSTRUCTIONS THEN
000217 001 C BRANCH TO APPROPRIATE SECTION OF CODE.
000218 001 280 GO TO 14
000219 001 IP=1
000220 001 IF (IVAR.FN.INST(1)) GO TO 290
000221 001 280 CONTINUE
000222 001 GO TO 270
000223 001 C
000224 001 290 GO TO (300+360+390+310+320+330+340+350+370+380+400+410+420+470+430)00290
000225 001 1+440+460+480+490), IP
000226 001 C
000227 001 C ENVIRONMENT - COASTAL OR DEEP
000228 001 300 NVIRON=1
000229 001 IF (Z1.FN.2.) NVIRON=2
000230 001 STEP(1)=22
000231 001 EVAL(1)=73
000232 001 GO TO 270
000233 001 C
000234 001 C WAVE HEIGHT
000235 001 310 WAVHT=71
000236 001 STEP(1)=22
000237 001 EVAL(1)=23
000238 001 GO TO 270
000239 001 C
000240 001 C BOTTOM ROUGHNESS
000241 001 320 HTMORF=71
000242 001 STEP(1)=22
000243 001 EVAL(1)=73
000244 001 GO TO 270
000245 001 C
000246 001 C LENGTH OF TARGET
000247 001 330 TARGLN=71
000248 001 STEP(1)=22
000249 001 EVAL(1)=23
000250 001 GO TO 270
000251 001 C
000252 001 C HEIGHT OF TARGET
000253 001 340 TARGHT=71
000254 001 STEP(1)=22

```

U00255	U01	EDVAL(7)=23	DAIN255
U00256	U01	GO TO 270	DAIN256
U00257	U01	C	DAIN257
U00258	U01	C SOURCE RECEIVER SEPARATION	DAIN258
U00259	U01	350 SRSEFP=1	DAIN259
U00260	U01	STEP(1)=2	DAIN260
U00261	U01	EDVAL(1)=23	DAIN261
U00262	U01	GO TO 270	DAIN262
U00263	U01	C	DAIN263
U00264	U01	C SOURCE TYPE	DAIN264
U00265	U01	360 ISTYPE=1	DAIN265
U00266	U01	IF (Z1.FU.2.) ISTYPE=2	DAIN266
U00267	U01	IF (Z1.FU.3.) ISTYPE=3	DAIN267
U00268	U01	STEP(2)=2	DAIN268
U00269	U01	EDVAL(2)=23	DAIN269
U00270	U01	GO TO 270	DAIN270
U00271	U01	C	DAIN271
U00272	U01	C SOURCE POWER	DAIN272
U00273	U01	370 SPWR=1	DAIN273
U00274	U01	STEP(3)=2	DAIN274
U00275	U01	EDVAL(4)=23	DAIN275
U00276	U01	GO TO 270	DAIN276
U00277	U01	C	DAIN277
U00278	U01	C FULL SOURCE ANGLE	DAIN278
U00279	U01	380 SDELF=1	DAIN279
U00280	U01	STEP(10)=2	DAIN280
U00281	U01	EDVAL(10)=23	DAIN281
U00282	U01	IF (SDELF.GT.180.) SDELF=SULELF-180.	DAIN282
U00283	U01	IF (ARS(2).EQ.0.) EDVAL(10)=SULELF	DAIN283
U00284	U01	GO TO 270	DAIN284
U00285	U01	C	DAIN285
U00286	U01	C RECEIVER TYPE	DAIN286
U00287	U01	390 IRTP=1	DAIN287
U00288	U01	IF (Z1.FU.2.) IRTP=2	DAIN288
U00289	U01	STEP(9)=2	DAIN289
U00290	U01	EDVAL(5)=23	DAIN290
U00291	U01	GO TO 270	DAIN291
U00292	U01	C	DAIN292
U00293	U01	C FULL RECEIVER ANGLE	DAIN293
U00294	U01	400 RDELF=1	DAIN294
U00295	U01	STEP(11)=2	DAIN295
U00296	U01	EDVAL(11)=23	DAIN296
U00297	U01	IF (RDELF.GT.180.) RDELF=RUELF-180.	DAIN297
U00298	U01	IF (ARS(2).EQ.0.) EDVAL(11)=RUELF	DAIN298
U00299	U01	GO TO 270	DAIN299
U00300	U01	C	DAIN300
U00301	U01	C TRANSMISSION OF OPTICS	DAIN301
U00302	U01	410 OPRAN=71	DAIN302
U00303	U01	STEP(12)=2	DAIN303
U00304	U01	EDVAL(12)=23	DAIN304
U00305	U01	IF (OPTRAN.GT.1.) OPRAN=0.	DAIN305
U00306	U01	IF (ARS(2).EQ.0.) EDVAL(12)=OPTRAN	DAIN306
U00307	U01	GO TO 270	DAIN307
U00308	U01	C	DAIN308
U00309	U01	C F-NUMFH	DAIN309
U00310	U01	420 FNUM=71	DAIN310
U00311	U01	STEP(13)=2	DAIN311
U00312	U01	EDVAL(13)=23	DAIN312
U00313	U01	IF (FNUM.LT..5) FNUM=.5	DAIN313
U00314	U01	IF (ARS(2).EQ.0.) EDVAL(13)=FNUM	DAIN314
U00315	U01	GO TO 270	DAIN315
U00316	U01	C	DAIN316
U00317	U01	C SET FLAG TO CALCULATE MAXIMUM WIDTH	DAIN317
U00318	U01	430 IF (ICAL.FH.2) GO TO 450	DAIN318
U00319	U01	ICAL=2	DAIN319
U00320	U01	GO TO 270	DAIN320
U00321	U01	C	DAIN321
U00322	U01	C SET FLAG TO CALCULATE MAXIMUM RANGE	DAIN322
U00323	U01	440 IF (ICAL.FH.3) GO TO 450	DAIN323
U00324	U01	ICAL=2	DAIN324
U00325	U01	GO TO 270	DAIN325
U00326	U01	C	DAIN326
U00327	U01	C SET FLAG TO CALCULATE BOTH MAXIMUM WIDTH AND	DAIN327
U00328	U01	C MAXIMUM RANGE	DAIN328
U00329	U01	450 ICAL=2	DAIN329
U00330	U01	GO TO 270	DAIN330
U00331	U01	C	DAIN331
U00332	U01	C SET FLAG TO CALCULATE SINGLE RANGE ONLY	DAIN332
U00333	U01	460 ICAL=1	DAIN333
U00334	U01	GO TO 270	DAIN334
U00335	U01	C	DAIN335
U00336	U01	C RANGE	DAIN336
U00337	U01	470 HEIGHT=71	DAIN337
U00338	U01	STEP(14)=2	DAIN338
U00339	U01	EDVAL(14)=23	DAIN339
U00340	U01	GO TO 270	DAIN340

000341	001	C		DAIN341
000342	001	C	SET FLAG TO PERFORM CALCULATIONS! OR	DAIN342
000343	001	C	HF-WHITE IS IN EFFECT - READ NEW SET OF CHANGES	DAIN343
000344	001	C	480 ISTART=1	DAIN344
000345	001	C		DAIN345
000346	001	C	490 CONTINUE	DAIN346
000347	001	C	HFURN	DAIN347
000348	001	C		DAIN348
000349	001	C		DAIN349
000350	001	C		DAIN350
000351	001	C	500 FORMAT (46H, IF HATCH RUN, ENTER - BATCH1 OTHERWISE RETURN)	DAIN351
000352	001	C	510 FORMAT (A6,3X,3F10.0)	DAIN352
000353	001	C	520 FORMAT (1X,A6,4H RUN)	DAIN353
000354	001	C	530 FORMAT (11,A1)	DAIN354
000355	001	C	540 FORMAT (6I10,0)	DAIN355
000356	001	C	550 FORMAT (1X,A6,3X,3E15.5)	DAIN356
000357	001	C	560 FORMAT (27H,2PHCONVENTIONAL TV SYSTEM/3Y+11ENVIRONMENT)	DAIN357
000358	001	C	570 FORMAT (26H, COASTAL OR DEEP, 1 OR 2:?)	DAIN358
000359	001	C	580 FORMAT (32H, WAVE HEIGHT-PEAK TO TROUGH,FT:?)	DAIN359
000360	001	C	590 FORMAT (37H, BOTTOM ROUGHNESS-PEAK TO TROUGH,FT:?)	DAIN360
000361	001	C	600 FORMAT (3X,17H,TARGET DIMENSIONS)	DAIN361
000362	001	C	610 FORMAT (22H, LENGTH OF TARGET,FT:?)	DAIN362
000363	001	C	620 FORMAT (22H, HEIGHT OF TARGET,FT:?)	DAIN363
000364	001	C	630 FORMAT (3X,24H,PLATFORM CHARACTERISTICS)	DAIN364
000365	001	C	640 FORMAT (32H, SOURCE RECEIVER SEPARATION,FT:?)	DAIN365
000366	001	C	650 FORMAT (3X,22H,SOURCE CHARACTERISTICS)	DAIN366
000367	001	C	660 FORMAT (6I10, INCANDESCENT, THALLIUM TOPIDE, OR MERCURY VAPOR, 1, 2	DAIN367
000368	001	C	10H, #?)	DAIN368
000369	001	C	670 FORMAT (21H, SOURCE POWER,WATTS:?)	DAIN369
000370	001	C	680 FORMAT (24H, FULL SOURCE ANGLE,DEG:?)	DAIN370
000371	001	C	690 FORMAT (3X,24H,RECEIVER CHARACTERISTICS)	DAIN371
000372	001	C	700 FORMAT (25H, VIDEODISC OR SITE, 1 OR 2:?)	DAIN372
000373	001	C	710 FORMAT (26H, FULL RECEIVER ANGLE,DEG:?)	DAIN373
000374	001	C	720 FORMAT (25H, TRANSMISSION OF OPTICS:?)	DAIN374
000375	001	C	730 FORMAT (11H, F-NUMBER:?)	DAIN375
000376	001	C	740 FORMAT (44H, PARAMETRIC OR NON-PARAMETRIC MODE, 1 OR 2:?)	DAIN376
000377	001	C	750 FORMAT (46H, WATER TYPE(1), SOURCE TYPE(2), RECEIVER TYPE(3)/52H WAVE/DAIN377	
000378	001	C	1, PEGHT(4), ROTOM ROUGHNESS(5), TARGET LENGTH(6)/51H TARGET HEIGHT(DAIN378	
000379	001	C	27) & R SEPARATION(8), SOURCE POWEN(9)/36H SOURCE ANGLE(10), RECEIVER(11)/8H DAIN379	
000380	001	C	3 ANGLE(11)/47H OPTICS TRANSMISSION(12), F-NUMBER(13), RANGE(14)/8H DAIN380	
000381	001	C	4INDEX:?)	DAIN381
000382	001	C	760 FORMAT (40H, INCREMENTAL VALUE (IN FLOATING POINT):?)	DAIN382
000383	001	C	770 FORMAT (35H, ENDING VALUE (IN FLOATING POINT):?)	DAIN383
000384	001	C	780 FORMAT (25H, PARAMETER INPUT COMPLETE:?)	DAIN384
000385	001	C	790 FORMAT (43H, DO YOU WANT ANOTHER RUN, YES(1) OR NO(2):?)	DAIN385
000386	001	C	800 FORMAT (39H, CHANGE PARAMETER(S), YES(1) OR NO(2):?)	DAIN386
000387	001	C	810 FORMAT (32H, MORE CHANGES, YES(1) OR NO(2):?)	DAIN387
000388	001	C	820 FORMAT (48H, PARAMETRIC OR NON-PARAMETRIC VARIABLE, 1 OR 2:?)	DAIN388
000389	001	C	830 FORMAT (14H, CHANGE INDEX:?)	DAIN389
000390	001	C	840 FORMAT (45H, MORE PARAMETRIC VARIABLES, YES(1) OR NO(2):?)	DAIN390
000391	001	C	850 FORMAT (54H, MAX. RANGE(2), MAX. WIDTH(3), BOTH(4) OR NEITHER(5):?)	DAIN391
000392	001	C	ENJ	DAIN392

000001	001	C	SUBROUTINE DATA01 (TSUM)	DAOT 1
000002	001	C		DAOT 2
000003	001	C	SUBROUTINE DATA01 OUTPUTS THE FOLLOWING VARIABLES	DAOT 3
000004	001	C	MAXIMUM RANGE, MAXIMUM SWATH WIDTH, CONTRAST AT CENTER,	DAOT 4
000005	001	C	CONTRAST AT EDGE, AVAILABLE LINES AT CENTER, AVAILABLE	DAOT 5
000006	001	C	LINES AT EDGE	DAOT 6
000007	001	C		DAOT 7
000008	001	C	COMMON BLOCK OF OPTIONS	DAOT 8
000009	001	C	COMMON /UP15/ IRUN,IPARAM,ICAL	DAOT 9
000010	001	C		DAOT 10
000011	001	C	COMMON BLOCK OF I/O VARIABLES	DAOT 11
000012	001	C	COMMON /10LIST/ NIVORN, ISPTP, INTPTP, WAVEHT, BTMRUF, TANGLN,	DAOT 12
000013	001	C	1, TADHT, SHSEP, SPWNP, SUELP, RUELP, UPTRAN, FNUM, HFIGHT,	DAOT 13
000014	001	C	2, RANGM, RANGE, RANGAL, RANGMX, SWTHL, SWTHR, SWTHH,	DAOT 14
000015	001	C	3, SWTHL, AVSWL, AVSWR, AVSW, LINEN, LINEUD, CTHCEN, CTHEDG,	DAOT 15
000016	001	C	4, NUFXWL, NUFXWR, LIMFLG	DAOT 16
000017	001	C		DAOT 17
000018	001	C	DIMENSION 1A(5)	DAOT 18
000019	001	C	DIMENSION /FRO(16)	DAOT 19
000020	001	C	REAL LINEN,LINEUD	DAOT 20
000021	001	C		DAOT 21
000022	001	C	DATA TA/0H,POWER, 6HR, ANG/6MS, ANG/SHRKSKTH, SHUISPLA/	DAOT 22
000023	001	C		DAOT 23
000024	001	C	EQUIVALENCE (ZERO(1),RANGMN)	DAOT 24
000025	001	C	IF (LTFLG.LT.1) GO TO 10	DAOT 25
000026	001	C	NDLYWL=5	DAOT 26
000027	001	C	NDLYWR=5	DAOT 27
000028	001	C		DAOT 28
000029	001	C	10 IF ((IRUN,EN,1,OR,IPARAM,EO,1)) GO TO 70	DAOT 29
000030	001	C		DAOT 30

```

000031 001      NAMF1=UHLOASTA          DAOT 31
000032 001      NAMF2=UHIL          DAOT 32
000033 001      IF (INVIPUN.EQ.1) GO TO 20  DAOT 33
000034 001      NAMF1=UHUFFP          DAOT 34
000035 001      NAMF2=UH          DAOT 35
000036 001      20 NAMF3=UHINLAND          DAOT 36
000037 001      NAMF4=UH.          DAOT 37
000038 001      IF (1S1PE-2) 50,30,40  DAOT 38
000039 001      30 NAMF3=UHITHAL.          DAOT 39
000040 001      NAMF4=UHIOU.          DAOT 40
000041 001      GO TO 50  DAOT 41
000042 001      40 NAMF3=UHMEHC.          DAOT 42
000043 001      NAMF4=UHVAF.          DAOT 43
000044 001      50 NAMF5=UHVIIICO          DAOT 44
000045 001      NAMF6=UHN          DAOT 45
000046 001      IF (1R1PE.EQ.1) GO TO 60  DAOT 46
000047 001      NAMF5=UHSIT          DAOT 47
000048 001      NAMF6=UH          DAOT 48
000049 001      C
000050 001      60 PRINT 40, NAMF1+NAMF2+NAMF3+NAMF4  DAOT 49
000051 001      PRINT 100, WAVEHT,SPONK  DAOT 50
000052 001      PRINT 110, RTMHUF,SNELF  DAOT 51
000053 001      PRINT 120, TARGLN,NAME5,NAME6  DAOT 52
000054 001      PRINT 130, TARGHT,SNELF  DAOT 53
000055 001      PRINT 140, OPTRAN  DAOT 54
000056 001      PRINT 150, SNSEP,FNUM  DAOT 55
000057 001      PRINT 160, RANGE,RANGAL  DAOT 56
000058 001      PRINT 170, LINCEN,LINEDG  DAOT 57
000059 001      PRINT 180, CTRCEN,CTRLEDG  DAOT 58
000060 001      PRINT 190, AVSWL,SWTHL  DAOT 59
000061 001      PRINT 200, TA(INDEXL)  DAOT 60
000062 001      PRINT 210, AVSWR,SWTHR  DAOT 61
000063 001      PRINT 220, TA(INDEXR)  DAOT 62
000064 001      PRINT 230, AVSW,SWTH  DAOT 63
000065 001      C
000066 001      70 IF (1SUM.EQ.1) REVINU 4  DAOT 64
000067 001      WRITE (9) NVIPUN,DAVENT,HTMRUF  DAOT 65
000068 001      WRITE (9) TARGLN,TARGHT  DAOT 66
000069 001      WRITE (9) LSSEP  DAOT 67
000070 001      WRITE (9) 1STPE,SPONK,SNELF  DAOT 68
000071 001      WRITE (9) 1RTPE,SNELF,OPTTRAN,FNUM  DAOT 69
000072 001      WRITE (9) RANGE,RANGAL,RANGMN,RANGMX,RANGE8  DAOT 70
000073 001      WRITE (9) LINCEN,LINEDG,CTRCCEN,CTRLEDG  DAOT 71
000074 001      WRITE (9) AVSWL,SWTHL,TA(INDEXL)  DAOT 72
000075 001      WRITE (9) AVSWR,SWTHR,TA(INDEXR)  DAOT 73
000076 001      WRITE (9) AVSW,SWTH,SWTHB  DAOT 74
000077 001      C
000078 001      C
000079 001      80 DO 1=1,16  DAOT 75
000080 001      ZEROIT)=0.  DAOT 76
000081 001      90 CONTINUE  DAOT 77
000082 001      NDC(WL)=0  DAOT 78
000083 001      NDE(YR)=0  DAOT 79
000084 001      LIMFLG=0  DAOT 80
000085 001      C
000086 001      RE1=MNN  DAOT 81
000087 001      C
000088 001      C
000089 001      C
000090 001      90 FORMAT (17X,22HCONVENTIONAL TV SYSTEM/13H WATER TYPE: ,12A6,14X,RHS) DAOT 90
000091 001      SOURCE: ,12A6)  DAOT 91
000092 001      100 FORMAT (11H WAVE HEIGHT,FT:,F12.5,11X,14H SOURCE POWER:,F12.5)  DAOT 92
000093 001      110 FORMAT (11H HOTTUV,FT:,F12.5,10X,12H SOURCE REAM:,F12.5)  DAOT 93
000094 001      120 FORMAT (11H LENGTH,FT:,F12.5,16X,9H RECEIVEVER:,2A6)  DAOT 94
000095 001      130 FORMAT (11H HEIGHT,FT:,F12.5,16X,14H HFCCIVER REAM:,F12.5)  DAOT 95
000096 001      140 FORMAT (13H TAU:,F12.5)  DAOT 96
000097 001      150 FORMAT (11H S-H, SEP:,F12.5,16X,3HF:,F12.5)  DAOT 97
000098 001      160 FORMAT (11H RANGE,FT:,F12.5,17X,9HRANGF:AL:,F12.5)  DAOT 98
000099 001      170 FORMAT (1/25H AVAILABLE LINES AT CTR:,F12.5,2X,1HAT EDGF:,F12.5)  DAOT 99
000100 001      180 FORMAT (20H CONTRAST AT CFNTER:,F12.5,7X,1HAT EDGE:,F12.5)  DAOT 100
000101 001      190 FORMAT (1/20H AVAILABLE L H S SW WIDTH:,F9.5,13H FT ACTUAL =,F9.5,DAOT101
000102 001      13H FT)  DAOT 102
000103 001      200 FORMAT (12H LIMITED BY:,A6)  DAOT 103
000104 001      210 FORMAT (1/20H AVAILABLE R H S SW WIDTH:,F9.5,13H FT ACTUAL =,F9.5,DAOT104
000105 001      13H FT)  DAOT 105
000106 001      220 FORMAT (1/20H AVAILABLE SW WIDTH:,F9.5,13H FT ACTUAL =,F9.5,3H FT) DAOT106
000107 001      ENU  DAOT 107-

```

BEST AVAILABLE COPY

```

000001 001      FUNCTION DFUNC(THET+SRSEP+RAN+EANG+TWAT)
000002 001      C
000003 001      C      COMMON BLOCK OF COMPUTING CONSTANTS
000004 001      COMMON /IVLCOM/ A(2*2), B(2*11*2), C(20), D(2*2), E(10),
000005 001      1 FM(2*2), G(2*2*3), H(2*2*3), N(10), O(15), P(2*2),
000006 001      2 Q(15), S(2), T(11), W(10), Z(2), PI
000007 001      C
000008 001      D1=ATAN((SRSEP+RAN+TAN(THET+EANG))/RAN)-EANG)
000009 001      D2=(O(TWAT+1)/COS(D1+EANG))+(1/COS(THET+EANG))
000010 001      D3=1.0*EXP(-D2*A(TWAT+1)+RAN)
000011 001      D4=(D3*(COS(EANG+D1)*3.0)*(COS(THET)*4.0))/(RAN+RAN)
000012 001      DFUNC=D4
000013 001      RETURN
000014 001      END

000001 001      FUNCTION EFUNC(CURR+J)
000002 001      C
000003 001      C      COMMON BLOCK OF COMPUTING CONSTANTS
000004 001      COMMON /IVLCOM/ A(2*2), B(2*11*2), C(20), D(2*2), E(10),
000005 001      1 FM(2*2), G(2*2*3), H(2*2*3), N(10), O(15), P(2*2),
000006 001      2 Q(15), S(2), T(11), W(10), Z(2), PI
000007 001      C
000008 001      EFUNC=(ALOG((CURR+Z(J))/FM(J+1))/FM(J+2))
000009 001      RETURN
000010 001      END

000001 001      SUBROUTINE FNK (II+IB)
000002 001      C
000003 001      C      COMMON BLOCK OF COMPUTING CONSTANTS
000004 001      COMMON /IVLCOM/ A(2*2), B(2*11*2), C(20), D(2*2), E(10),
000005 001      1 FM(2*2), G(2*2*3), H(2*2*3), N(10), O(15), P(2*2),
000006 001      2 Q(15), S(2), T(11), W(10), Z(2), PI
000007 001      C
000008 001      ID2=1
000009 001      DO 10 ID1=1,I+IB
000010 001      E1(D1)=T02
000011 001      10 ID2=ID2+1
000012 001      IR1=IR-1
000013 001      DO 20 ID1=1,I+IB1
000014 001      IF (W(ID1).GT.W(ID1+1)) GO TO 20
000015 001      W(ID1+1)=W(ID1)
000016 001      E(ID1+1)=E(ID1)
000017 001      20 CONTINUE
000018 001      RETURN
000019 001      END

000001 001      FUNCTION GFUNC(A+1,RAN,ANGLE+EANG+SRSEP)
000002 001      C
000003 001      C      CALCULATE RMSCTN WIDTH
000004 001      C
000005 001      D1=-(ALOG(R/A))/(RAN*TAN(ANGLE+EANG)-(SRSEP/2.0))
000006 001      GFUNC=-(ALOG(.07/A))/D1
000007 001      RETURN
000008 001      END

000001 001      SUBROUTINE HLIMIT (TFLAG)
000002 001      C
000003 001      C      SUBROUTINE HLIMIT COMPUTES THE HEIGHT LIMITS
000004 001      C
000005 001      C      COMMON BLOCK OF OPTIONS
000006 001      COMMON /OPTS/ IRUN,IPARAM,ICAL
000007 001      C
000008 001      C      COMMON BLOCK OF I/O VARIABLES
000009 001      COMMON /IOIST/ NIVIRON, ISIP, ITMPH, WAVEHT, BTMRUF, TARGLN,
000010 001      1 TAIGHT, SWSEP, SPWNU, SULFL, RUEFL, OPTRAN, FNUM, HFIGHT,
000011 001      2 RANGMN, RANGE, HANGAL, HANGER, HANGMX, SWTHL, SWTHR, SWTHH,
000012 001      3 SWTHH, AVSLW, AVSWR, AVSW, LINLEN, LINEUG, CTHCEN, CTHEDG,
000013 001      4 NIFXWL, NIFXWW, LIMFLG
000014 001      C
000015 001      C      COMMON BLOCK OF COMPUTING CONSTANTS
000016 001      COMMON /IVLCOM/ A(2*2), B(2*11*2), C(20), D(2*2), E(10),
000017 001      1 FM(2*2), G(2*2*3), H(2*2*3), N(10), O(15), P(2*2),
000018 001      2 Q(15), S(2), T(11), W(10), Z(2), PI
000019 001      C
000020 001      REAL LINLEN,LINEUG
000021 001      C
000022 001      C      REQUIRED DEPTH OF FIELD, FEET
000023 001      D1FLD=.5*(WAVEHT+BTMRUF)+TAIGHT
000024 001      C
000025 001      C      INITIAL ESTIMATE MINIMUM RANGE, FEET
000026 001      H1=(.05/A(NIVIRON+1))+3.281

```

```

000027 001      ANGL ER= .00072665*RDPLF          HLIM 27
000028 001      ANGL ESZ=.00072665*SUPLF          HLIM 28
000029 001      10 EX=ATAN(SRSEP/(2.*R1))
000030 001      IF (AMAX1(X+ANGLEH,EX+ANGLES).LT.(PII+.49)) GO TO 20  HLIM 29
000031 001      R1=1.05*R1
000032 001      GO TO 10
000033 001      C
000034 001      20 T1=DFIELDU+(SRSEP/(TAN(ANGLES)+TAN(ANGLEL)))
000035 001      30 EX=ATAN(SRSEP/(2.*R1))
000036 001      T2=DFIELDU+(SRSEP/(TAN(ANGLES+EX)+TAN(ANGLEL+EX)))
000037 001      IF (T2.GT.R1) GO TO 40
000038 001      T2=0.
000039 001      GO TO 50
000040 001      C
000041 001      40 IF (ARS(11-T2).LT..001) GO TO 50
000042 001      T1=T2
000043 001      GO TO 30
000044 001      C
000045 001      50 SET MINIMUM RANGE: FFET          HLIM 44
000046 001      50 RANGMN=AMAX1(T2,R1)          HLIM 45
000047 001      RANGMN=AMAX1(RANGMN,SRSEP)          HLIM 46
000048 001      C
000049 001      C
000050 001      T1=RUELF*.5          HLIM 49
000051 001      T2=RUELF*.5          HLIM 50
000052 001      DINVIRON,1)=CFUNC(1,T1,NVIRON)          HLIM 51
000053 001      DINVIRON,2)=CFUNC(2,T2,NVIRON)          HLIM 52
000054 001      C
000055 001      IF (IPUN,EU,1) GO TO 70
000056 001      IF (IFLAG,NE,0) GO TO 70
000057 001      PRINT 130, RANGMN          HLIM 56
000058 001      IF (IPARAM,NE,1) GO TO 60
000059 001      PRINT 160
000060 001      READ (5,170) HEIGHT
000061 001      GO TO 70
000062 001      60 PRINT 140
000063 001      READ (5,150) ICAL
000064 001      TU CONTINUE
000065 001      IF (ICAL,EU,4) GO TO 90
000066 001      IF (ICAL-2) 100,90,90
000067 001      C
000068 001      C      GET MAXIMUM SWATH WIDTH          HLIM 67
000069 001      90 CONTINUE
000070 001      CALI TWFUNL (-1.,.3)          HLIM 68
000071 001      RANGERZ=0(1)*3.281          HLIM 69
000072 001      SWATHZ=0(1)*3.281          HLIM 70
000073 001      IF (IPARAM,EU,1) GO TO 110
000074 001      GO TO 120
000075 001      C
000076 001      C      GET MAXIMUM RANGE          HLIM 75
000077 001      90 CONTINUE
000078 001      CALI TWFUNL (-1.,.2)          HLIM 76
000079 001      RANGMX=0(1)*3.281          HLIM 77
000080 001      IF (ICAL,EU,4) GO TO 80
000081 001      IF (IPARAM,EU,1) GO TO 110
000082 001      GO TO 120
000083 001      C
000084 001      110 IF (IRUN,EU,1.OR.(IPARAM,EU,1)) GO TO 110
000085 001      PRINT 160
000086 001      READ (5,170) HEIGHT
000087 001      C
000088 001      C
000089 001      C      CALCULATE SINGLE RANGE          HLIM 88
000090 001      110 CONTINUE
000091 001      IF (HEIGHT,EU,0) GO TO 120
000092 001      CALI TWFUNL (HEIGHT,1)
000093 001      120 CONTINUE
000094 001      RANGER=0(1)*3.281          HLIM 89
000095 001      RANGAL=0(1)*0(NVIRON,1)          HLIM 90
000096 001      AVSWL=RANGE*TAN(ANGLEL-G(13))+0.5*SRSEP          HLIM 91
000097 001      SWATHL=0(1)*3.281          HLIM 92
000098 001      AVSWH=RANGE*TAN(ANGLEH-G(13))-0.5*SRSEP          HLIM 93
000099 001      SWATHZ=0(1)*3.281          HLIM 94
000100 001      AVSW4=AVSWL+AVSWR          HLIM 95
000101 001      SWATH=0(1)*3.281          HLIM 96
000102 001      C
000103 001      C      DISPLAY LIMIT IN LIMIT FLAG.          HLIM 103
000104 001      C      STORE RIGHT AND LEFT-HAND SIDE WIDTH INDICES: AVERAGE          HLIM 104
000105 001      C      LIMFLG=0(3)          HLIM 105
000106 001      LIMFLG=0(3)
000107 001      NDYWL=0(9)          HLIM 106
000108 001      NDYWR=0(11)          HLIM 107
000109 001      LINPEN=0(3)          HLIM 108
000110 001      LINFLG=0(4)          HLIM 109
000111 001      CTHFLG=0(6)          HLIM 110
000112 001      CTHFLG=0(7)          HLIM 111

```

U00113	U01	C	HLIM113
U00114	U01	RETURNN	HLIM114
U00115	U01	C	HLIM115
U00116	U01	C	HLIM116
U00117	U01	C	HLIM117
U00118	U01	130 FORMAT (17H MINIMUM RANGE IS: F9.5,5H FEET)	HLIM118
U00119	U01	140 FORMAT (46H ONE RANGE(1),MAX. RANGE(2) OR MAX. WIDTH(3):?)	HLIM119
U00120	U01	150 FORMAT (1I)	HLIM120
U00121	U01	160 FORMAT (1IH RANGE,FT:?)	HLIM121
U00122	U01	170 FORMAT (G10.0)	HLIM122
U00123	U01	END	HLIM123-

***** MAIN *****

```

000001 001 C INPUT FILES:
000002 001 C   UNIT 5 (CARD READER) -
000003 001 C   DATA OF VARIABLE INITIALIZATION AS FOLLOWS:
000004 001 C   ENVIRONMENT, TARGET DIMENSIONS, PLATFORM CHARACTER-
000005 001 C  ISTICS, SOURCE CHARACTERISTICS, AND RECEIVER
000006 001 C   CHARACTERISTICS.
000007 001 C OUTPUT FILES:
000008 001 C   UNIT 6 (PRINTER)
000009 001 C   INTERMEDIATE AND SUMMARY INFORMATION
000010 001 C
000011 001 C   COMMON BLOCK OF COMPUTING CONSTANTS
000012 001 C   COMMON /IVCOM/ A(2,2), B(2,11,2), C(20), D(2,2), E(10),
000013 001 C   1, FM(2,2), G(2,2,3), H(2,2,3), N(10), O(15), P(2,2),
000014 001 C   2, Q(15), S(2), T(11), W(10), Z(2), PI
000015 001 C
000016 001 C   COMMON BLOCK OF OPTIONS
000017 001 C   COMMON /OPTS/ IRUN, IPARAM, ICAL
000018 001 C
000019 001 C   ATTN. CONST.
000020 001 C   DATA ((AT(I,J),J=1,2),I=1,2) / .252, .273, .049, .049 /
000021 001 C
000022 001 C   EFFECTIVE ALFA ARRAY
000023 001 C   DATA ((AL(I,J,K),I=1,2),J=1,11),K=1,2) / 1., 1., .87, .91, .69,
000024 001 C   1./.49, .62, .72, .49, .63, .58, .55, .33, .51, .29, .49, .24, .46,
000025 001 C   2./.20, .42, .17, .41, 1., 1., .87, .41, .68, .78, .59, .72, .45,
000026 001 C   3./.31, .74, .55, .24, .51, .25, .49, .34, .44, .15, .42, .14, .41 /
000027 001 C
000028 001 C
000029 001 C   NUMBER OF LINES MATRIX
000030 001 C   DATA ((FM(I,J),J=1,2),I=1,2) / .000127, .00863, 9.51E-14, .0133 /
000031 001 C
000032 001 C   SOURCE RECEIVER MATRIX
000033 001 C   DATA ((G(I,J,K),I=1,2),J=1,2),K=1,3) / .00361, .00106, .002,
000034 001 C   1./.00057, .0136, .00325, .001747, .00304, .00610, .00287, .00292,
000035 001 C   2./.00346 /
000036 001 C
000037 001 C   SOURCE RECEIVER MATRIX (BACKSCATTER)
000038 001 C   DATA ((H(I,J,K),I=1,2),J=1,2),K=1,3) / .000435, .00281, .00277,
000039 001 C   1./.00219, .0162, .0134, .00922, .00924, .0090, .00430, .00642,
000040 001 C   2./.00574 /
000041 001 C
000042 001 C   LAG MATRIX
000043 001 C   DATA ((P(I,J),J=1,2),I=1,2) / 3.70, 105.0, 5.79E-3, 169.0 /
000044 001 C
000045 001 C   SCATTERING CONSTANT
000046 001 C   DATA S / .239, .030 /
000047 001 C
000048 001 C   ANGLES
000049 001 C   DATA T / 0.0, .05, .5, 1., 2.5, 5., 7.5, 10., 20., 30., 40. /
000050 001 C
000051 001 C   RECEIVER FACTOR
000052 001 C   DATA Z / 1070000.0, 1. /
000053 001 C
000054 001 C   PI
000055 001 C   DATA PI / 3.14159265 /
000056 001 C
000057 001 C   INITIALIZE THE NUMBER OF SUMMARIES COUNTERS AND SET THE
000058 001 C   EXECUTION-BRANCH FLAG FOR READING THE FIRST CARD OF INPUT.
000059 001 C   THIS CARD SPECIFIES THE TYPE OF PROCESSING: BATCH OR DEMAND
000060 001 C   TERMINAL.
000061 001 C   ISSTART=-1
000062 001 C   ISUM=0
000063 001 C
000064 001 C   READ DATA INPUT. IF NO DATA IS GIVEN, STOP EXECUTION.
000065 001 C   IN CALL DATAIN (ISSTART,ISUM)
000066 001 C   IFLAG=0

```

```

U00067 001 IF (ISTART.NE.1) GO TO 30
U00068 001 C
U00069 001 C
U00070 001 C
U00071 001 C
U00072 001 C
U00073 001 C
U00074 001 C
U00075 001 C
U00076 001 C
U00077 001 C
U00078 001 C
U00079 001 C
U00080 001 C
U00081 001 C
U00082 001 C
U00083 001 C
U00084 001 C
U00085 001 C
U00086 001 C
U00087 001 C
U00088 001 C
U00089 001 C
U00090 001 C
U00091 001 C
U00092 001 C

      IF (ISTART.NE.1) GO TO 30
      INCREMENT THE NUMBER OF SUMMARIES COUNTER. PERFORM
      CALCULATIONS AND OUTPUT RESULTS.
      20 ISUMTISUM1
      CALI HLIMIT (IFLAG)
      CALI DATAOT (ISUM)
      RE-SET THE EXECUTION-HRANCH FLAG TO STOP EXECUTION. IF
      IN PARAMETRIC MODE, UPDATE THE NECESSARY PARAMETER. (IF THE
      VALUE OF A PARAMETER HAS BEEN CHANGED, SUBROUTINE UPDATE WILL
      SET THE EXECUTION-HRANCH FLAG TO PERFORM NEW CALCULATIONS.)
      IF IN NON-PARAMETRIC MODE, OR IF NO CHANGE IN THE PARAMETERS
      IS MADE, CONTINUE HEADING INPUT.
      ISTARTN
      IF (IPARAM.EQ.1) CALL UPDATE (ISTART)
      IFLAG=ISTART
      IF (ISTART.EQ.1) GO TO 20
      GO TO 10
      IF SUMMARY INFORMATION IS GIVEN, OUTPUT THE SUMMARY
      TABLE. STOP EXECUTION.
      30 IF (ISUM.NE.0) CALL SUMMARY (IRUN,ISUM)
      STOP
      ENU

```

```

U00001 001 SUBROUTINE SUMMARY (IRUN,ISUM)
U00002 001 C
U00003 001 C
U00004 001 C
U00005 001 C
U00006 001 DIMENSION NVIIRON(8),WAVEHT(8),RTMHRIF(8),TARGLN(8),TARGHT(8),
U00007 001 1 SRSPF(8),TSPTE(8),SPWRK(8),SDELF(8),IRPTE(8),
U00008 001 2 RDELF(8),OPTNAN(8),FNUM(8),AVSWH(8),AVSW(8),RANGE(8),
U00009 001 3 RANGMN(8),RANGMX(8),RANGE(8),SWWTH(8),SWWTHL(8),SWWTHR(8),
U00010 001 4 SWWTHN(8),RANGAL(8),LINCEN(8),LINEDG(8),CTRLNG(8),CTREDG(8),
U00011 001 5 AVSWL(8),IAL(8),IAK(8),IPART2(8)
U00012 001 UTMENSION NAME(14)
U00013 001 REAI LINLEN+LINEFG
U00014 001 C
U00015 001 DATA NAME /6H COA,4HSTAL,6H ,4HDFEP,
U00016 001 1 6H INC,4HANL,4HTHAL,4HIND,4H
U00017 001 2 6HMRCC,4HVAP,6H VID,4HICON,
U00018 001 3 6H ,4HSIT/
U00019 001 REAIND 9
U00020 001 C
U00021 001 C
U00022 001 ICK=4
U00023 001 IF (IRUN.EQ.1) ICK=1
U00024 001 KNPEN
U00025 001 DO 70 I=1,ISUM
U00026 001 KNTREKRNTH+1
U00027 001 READ (9) NVIIRON(KNTR),WAVEHT(KNTR),RTMHRIF(KNTR)
U00028 001 READ (9) TARGLN(KNTR),TARGHT(KNTR)
U00029 001 READ (9) SRSPF(KNTR)
U00030 001 READ (9) TSPTE(KNTR),SPWRK(KNTR),SDELF(KNTR)
U00031 001 READ (9) IRPTE(KNTR),RUFLE(KNTR),OPTNAN(KNTR),FNUM(KNTR)
U00032 001 READ (9) RANGE(KNTR),RANGAL(KNTR),RANGMN(KNTR),RANGMX(KNTR),RANGE
U00033 001 1(KNTR)
U00034 001 READ (9) LINCEN(KNTR),LINEDG(KNTR),CTRCEN(KNTR),CTREDG(KNTR)
U00035 001 READ (9) AVSWL(KNTR),SWWTHL(KNTR),IAL(KNTR)
U00036 001 READ (9) AVSWR(KNTR),SWWTHR(KNTR),IAK(KNTR)
U00037 001 READ (9) AVSW(KNTR),SWWTH(KNTR),SWWTHR(KNTR)
U00038 001 C
U00039 001 IF (KNTR.EQ.ICK.OR.I.EQ.ISUM) GO TO 10
U00040 001 GO TO 10
U00041 001 C
U00042 001 10 CONTINUE
U00043 001 IF (IRUN.EQ.1) GO TO 20
U00044 001 PRINT 60
U00045 001 READ 15,YN
U00046 001 GO TO 30
U00047 001 20 PRINT 100
U00048 001 30 CONTINUE
U00049 001 C
U00050 001 DO 40 K=1,KNTR
U00051 001 IN=NVIRON(K)+NVIRON(K)-1
U00052 001 INP1=IN+1
U00053 001 NVIRON(K)=NAME(IN)
U00054 001 IPART2(K)=NAME(INP1)
U00055 001 40 CONTINUE

```

```

MAIN 67
MAIN 68
MAIN 69
MAIN 70
MAIN 71
MAIN 72
MAIN 73
MAIN 74
MAIN 75
MAIN 76
MAIN 77
MAIN 78
MAIN 79
MAIN 80
MAIN 81
MAIN 82
MAIN 83
MAIN 84
MAIN 85
MAIN 86
MAIN 87
MAIN 88
MAIN 89
MAIN 90
MAIN 91
MAIN 92-

```

```

SIIMA 1
SIIMA 2
SIIMA 3
SIIMA 4
SIIMA 5
SIIMA 6
SIIMA 7
SIIMA 8
SIIMA 9
SIIMA 10
SIIMA 11
SIIMA 12
SIIMA 13
SIIMA 14
SIIMA 15
SIIMA 16
SIIMA 17
SIIMA 18
SIIMA 19
SIIMA 20
SIIMA 21
SIIMA 22
SIIMA 23
SIIMA 24
SIIMA 25
SIIMA 26
SIIMA 27
SIIMA 28
SIIMA 29
SIIMA 30
SIIMA 31
SIIMA 32
SIIMA 33
SIIMA 34
SIIMA 35
SIIMA 36
SIIMA 37
SIIMA 38
SIIMA 39
SIIMA 40
SIIMA 41
SIIMA 42
SIIMA 43
SIIMA 44
SIIMA 45
SIIMA 46
SIIMA 47
SIIMA 48
SIIMA 49
SIIMA 50
SIIMA 51
SIIMA 52
SIIMA 53
SIIMA 54
SIIMA 55

```

```

000056 001 PRINT 150, ((INVIDON(K)+IPART2(K)),K=1,KNTR) SUMA 56
000057 001 PRINT 120, (WAVEHT(K),K=1,KNTR) SUMA 57
000058 001 PRINT 130, (BTMROW(K),K=1,KNTR) SUMA 58
000059 001 PRINT 140, (TARGLN(K),K=1,KNTR) SUMA 59
000060 001 PRINT 150, (TANGLT(K),K=1,KNTR) SUMA 60
000061 001 PRINT 160, (SRSPF(K),K=1,KNTR) SUMA 61
000062 001 DO 50 K=1,KNTR SUMA 62
000063 001 IN1$=T$F(K)+1$PE(K)+3 SUMA 63
000064 001 IN1$=IN1$1 SUMA 64
000065 001 IS1$E(K)=NAME(IN) SUMA 65
000066 001 IPART2(K)=NAME(INP1) SUMA 66
000067 001 50 CONTINUE SUMA 67
000068 001 PRINT 170, ((IS1$E(K)+IPART2(K)),K=1,KNTR) SUMA 68
000069 001 PRINT 180, (SPWNK(K),K=1,KNTR) SUMA 69
000070 001 PRINT 190, (SDELF(K),K=1,KNTR) SUMA 70
000071 001 DO 60 K=1,KNTR SUMA 71
000072 001 IN1$=T$F(K)+1$PE(K)+9 SUMA 72
000073 001 IN1$=IN1$1 SUMA 73
000074 001 IS1$E(K)=NAME(IN) SUMA 74
000075 001 IPART2(K)=NAME(INP1) SUMA 75
000076 001 60 CONTINUE SUMA 76
000077 001 PRINT 200, ((IN1$E(K)+IPART2(K)),K=1,KNTR) SUMA 77
000078 001 PRINT 210, (RDELF(K),K=1,KNTR) SUMA 78
000079 001 PRINT 220, (OPTRANK(K),K=1,KNTR) SUMA 79
000080 001 PRINT 230, (ENOM(K),K=1,KNTR) SUMA 80
000081 001 PRINT 240, (RANGE(R),K=1,KNTR) SUMA 81
000082 001 PRINT 250, (RANGAL(K),K=1,KNTR) SUMA 82
000083 001 PRINT 260, (RANGMN(K),K=1,KNTR) SUMA 83
000084 001 PRINT 270, (RANGMX(K),K=1,KNTR) SUMA 84
000085 001 PRINT 280, (RANGE4(K),K=1,KNTR) SUMA 85
000086 001 PRINT 290, (LINEC1(K),K=1,KNTR) SUMA 86
000087 001 PRINT 300, (LINFDS(K),K=1,KNTR) SUMA 87
000088 001 PRINT 310, (CTHCEN(K),K=1,KNTR) SUMA 88
000089 001 PRINT 320, (CTRFOG(K),K=1,KNTR) SUMA 89
000090 001 PRINT 330, (AVSML((K),K=1,KNTR) SUMA 90
000091 001 PRINT 340, (SWWTHL(K),K=1,KNTR) SUMA 91
000092 001 PRINT 340, (IAL(K),K=1,KNTR) SUMA 92
000093 001 PRINT 340, (AVSWK(K),K=1,KNTR) SUMA 93
000094 001 PRINT 340, (SWWTHR(K),K=1,KNTR) SUMA 94
000095 001 PRINT 340, (IAK(K),K=1,KNTR) SUMA 95
000096 001 PRINT 360, (AVSWK(K),K=1,KNTR) SUMA 96
000097 001 PRINT 370, (SWWTH(K),K=1,KNTR) SUMA 97
000098 001 PRINT 280, (SWWTHB(K),K=1,KNTR) SUMA 98
000099 001 C SUMA 99
000100 001 KNTR=0 SUMA100
000101 001 70 CONTINUE SUMA101
000102 001 ISUMEN SUMA102
000103 001 C SUMA103
000104 001 RFLURN SUMA104
000105 001 C SUMA105
000106 001 C SUMA106
000107 001 C SUMA107
000108 001 80 FORMAT (24H CLEAR SCREEN AND RETURN) SUMA108
000109 001 90 FORMAT (80A1) SUMA109
000110 001 100 FORMAT (1H1) SUMA110
000111 001 110 FORMAT (12H WAFER TYPE:,11X,8(2X,A6,A4)) SUMA111
000112 001 120 FORMAT (16H WAVE HEIGHT,FT:,7X,AF12,4) SUMA112
000113 001 130 FORMAT (21H BOTTOM ROUGHNESS,FT:,2X,8F12,4) SUMA113
000114 001 140 FORMAT (18H TARGET LENGTH,FT:,5X,AF12,4) SUMA114
000115 001 150 FORMAT (18H TARGET HEIGHT,FT:,5X,AF12,4) SUMA115
000116 001 160 FORMAT (15H S+R+SEP.,FT:,10X,AF12,4) SUMA116
000117 001 170 FORMAT (8H SOURCE:,15X,A(2X,A6,A4)) SUMA117
000118 001 180 FORMAT (12H SOURCE POWER,WATTS:,3X,AF12,4) SUMA118
000119 001 190 FORMAT (17H SOURCE PEAK,UFQ:,6X,8F12,4) SUMA119
000120 001 200 FORMAT (10H REFLIVER:,15X,8(2X,A6,A4)) SUMA120
000121 001 210 FORMAT (19H REFLIVER,UFQ:,4X,8F12,4) SUMA121
000122 001 220 FORMAT (5H TAU:,1H,X,AF12,4) SUMA122
000123 001 230 FORMAT (10H F-NIMHEP:,13X,AF12,4/) SUMA123
000124 001 240 FORMAT (10H RANGE,FT:,13X,AF12,4) SUMA124
000125 001 250 FORMAT (2X,10H RANGF,AL:,11X,AF12,4) SUMA125
000126 001 260 FORMAT (2X,12H MINIMUM,FT:,9X,AF12,4) SUMA126
000127 001 270 FORMAT (2X,12H MAXIMUM,FT:,9X,AF12,4) SUMA127
000128 001 280 FORMAT (2X,9H HEST,FT:,12X,AF12,4) SUMA128
000129 001 290 FORMAT (17H AV LINES AT CTR:,6X,8F12,4) SUMA129
000130 001 300 FORMAT (2X,9H AT EDGE:,12X,AF12,4) SUMA130
000131 001 310 FORMAT (17H CONTRAST AT CTR:,6X,8F12,4) SUMA131
000132 001 320 FORMAT (23H AV L.H.S. SW WIDTH,FT:,AF12,4) SUMA132
000133 001 330 FORMAT (2X,11H ACTUAL,FT:,10X,8F12,4) SUMA133
000134 001 340 FORMAT (2X,14H - LIMITED BY:,7X,8(6X,A6)) SUMA134
000135 001 350 FORMAT (23H AV R.H.S. SW WIDTH,FT:,AF12,4) SUMA135
000136 001 360 FORMAT (19H AV SWATH WIDTH,FT:,4X,8F12,4) SUMA136
000137 001 ENU SUMA137-

```

```

000001 001      SUBROUTINE TVFNC (RSET,ICAL)          TVFU 1
000002 001      C                                     TVFU 2
000003 001      C                                     TVFU 3
000004 001      C                                     TVFU 4
000005 001      COMMON /IOLIST/ NIVRON, ISIPE, INTPE, WAVEHT, BTMRF, TARGLN, TVFU 5
000006 001      1 TARGHT, SHSEP, SPOW, SUELF, RUELF, OPTRAN, FNUM, HIGHT, TVFU 6
000007 001      2 RANGMN, RANGE, RANGAL, RANGER, RANGMX, SWTHL, SWTHR, SWTHRA, TVFU 7
000008 001      3 SWTHB, AVSWL, AVSWR, AVSW, LINCEN, LINEUG, CTRCEN, CTREUG, TVFU 8
000009 001      4 NUFXWL, NUFXWR, LIMFLG               TVFU 9
000010 001      C                                     TVFU 10
000011 001      COMMON /COMC/ A(2,2), B(2,11,2), C(20), D(2,2), E(10), TVFU 11
000012 001      1 FM(2,2), G(2,2,3), H(2,2,3), NI(10), O(15), P(2,2), TVFU 12
000013 001      2 Q(15), S(2), T(11), W(10), Z(2), PII               TVFU 13
000014 001      C                                     TVFU 14
000015 001      C                                     TVFU 15
000016 001      HEAT LI NCEN,LINEUG               TVFU 16
000017 001      C                                     TVFU 17
000018 001      IENVIRON                         TVFU 18
000019 001      JSETPDE                           TVFU 19
000020 001      K=ISIPE                           TVFU 20
000021 001      C                                     TVFU 21
000022 001      C                                     TVFU 22
000023 001      C                                     TVFU 23
000024 001      UFE=.3048*(.5*(WAVEHT+BTMRF)+TARGHT)           TVFU 24
000025 001      ANGLEP=U0H72665*RUELF               TVFU 25
000026 001      ANGLEP=U0H72665*SUELF               TVFU 26
000027 001      SRSEPH1=.3048*SHSEP               TVFU 27
000028 001      SRSEPH2=.5*SRSEPH1               TVFU 28
000029 001      C                                     TVFU 29
000030 001      C                                     TVFU 30
000031 001      C                                     TVFU 31
000032 001      C                                     TVFU 32
000033 001      C                                     TVFU 33
000034 001      C                                     TVFU 34
000035 001      C                                     TVFU 35
000036 001      DO 10 IT=1,10                         TVFU 36
000037 001      NI(IT)=0                           TVFU 37
000038 001      Q(IT)=0                           TVFU 38
000039 001      U(IT)=0                           TVFU 39
000040 001      10 CONTINUE                         TVFU 40
000041 001      DO 20 IT=11,15                         TVFU 41
000042 001      Q(IT)=0                           TVFU 42
000043 001      O(IT)=0                           TVFU 43
000044 001      20 CONTINUE                         TVFU 44
000045 001      C                                     TVFU 45
000046 001      Q(2)=ANGLEP                         TVFU 46
000047 001      O(2)=O(2)                           TVFU 47
000048 001      IFLAG2=0                          TVFU 48
000049 001      IFLAG6=0                          TVFU 49
000050 001      N(1)=ICAL                           TVFU 50
000051 001      C                                     TVFU 51
000052 001      C                                     TVFU 52
000053 001      RANGE=.3048*RSET                         TVFU 53
000054 001      IF (ICAL.FG.1) GO TO 30               TVFU 54
000055 001      RANGE=.3048*RANGMN                         TVFU 55
000056 001      RANG=RANG1                         TVFU 56
000057 001      30 C(4)=1./A(1,1)                         TVFU 57
000058 001      40 U(1)=RANG                         TVFU 58
000059 001      EANG=TA1*(SRSEPH1/(RANG+RANG))           TVFU 59
000060 001      DELTA1=TAN(ANGLEP-EANG)                 TVFU 60
000061 001      DELTA2=TAN(ANGLEP+EANG)                 TVFU 61
000062 001      C(5)=A.*RANG*(DELTAT1+DELTAT2)           TVFU 62
000063 001      C(5)=C(5)/(.3048*TARGLN)                 TVFU 63
000064 001      IF ((C(5).LT.525.) GO TO 50               TVFU 64
000065 001      NI(3)=1                           TVFU 65
000066 001      U(1)=0                           TVFU 66
000067 001      GO TO 150                         TVFU 67
000068 001      50 NI(4)=1                           TVFU 68
000069 001      NI(3)=0                           TVFU 69
000070 001      T1=0                           TVFU 70
000071 001      T2=0                           TVFU 71
000072 001      C                                     TVFU 72
000073 001      C                                     TVFU 73
000074 001      C(6)=FFINC(10.,SRSEPH1*RANG+EANG,NIVRON)   TVFU 74
000075 001      TS=FFINC(C(6)*INTPE)                  TVFU 75
000076 001      O(3)=AMIN1((TS*A.)/C(5),4200./C(5))       TVFU 76
000077 001      TS=ANGLEP                         TVFU 77
000078 001      C(7)=FFINC(TS*SRSEPH1*RANG+EANG,NIVRON)   TVFU 78
000079 001      TS=FFINC(C(7)*INTPE)                  TVFU 79
000080 001      O(4)=AMIN1((TS*A.)/C(5),4200./C(5))       TVFU 80
000081 001      C                                     TVFU 81
000082 001      IF ((TS.GT.C(5)) GO TO 110               TVFU 82
000083 001      T1=1                           TVFU 83
000084 001      C                                     TVFU 84
000085 001      C                                     TVFU 85

```

0000046	001	T5=ANGLE	TVFU 86
0000047	001	60 T6=FUNC(T5+SRSEPI+RANG+EANG+NVIKON)	TVFU 87
0000048	001	T6=FUNC(T6+IRPE)	TVFU 88
0000049	001	IF (T6.GT.1.5) GO TO 70	TVFU 89
0000040	001	T3=1.	TVFU 90
0000041	001	GO TO 80	TVFU 91
0000042	001	C	TVFU 92
0000043	001	SIGNAL LIMITED?	TVFU 93
0000044	001	70 T3=0.	TVFU 94
0000045	001	80 IF (T3.LT.-.5) GO TO 100	TVFU 95
0000046	001	T1=T3	TVFU 96
0000047	001	T5=T5+.05*(ANGLEFR+EANG)	TVFU 97
0000048	001	IF (ARS(.15-EANG).LT.(-.5+EANG)) GO TO 90	TVFU 98
0000049	001	GO TO 60	TVFU 99
000100	001	C	TVFU100
000101	001	C ZERO SWATH WIDTH DUE TO POWER LIMIT	TVFU101
000102	001	90 U(5)=1.	TVFU102
000103	001	W(1)=SRSEP2	TVFU103
000104	001	W(4)=SRSEPI2	TVFU104
000105	001	U(2)=0.	TVFU105
000106	001	GO TO 120	TVFU106
000107	001	C	TVFU107
000108	001	C NORMAL EXIT FROM POWER LIMIT	TVFU108
000109	001	100 U(5)=1.	TVFU109
000110	001	U(2)=T5-1.05*(ANGLEFR+EANG))	TVFU110
000111	001	T4=ANGU+TAN(EANG-T5)	TVFU111
000112	001	W(1)=T4+SRSEP2	TVFU112
000113	001	W(4)=T4-SRSEPI2	TVFU113
000114	001	GO TO 120	TVFU114
000115	001	C	TVFU115
000116	001	C MAXIMUM WIDTH	TVFU116
000117	001	110 W(1)=RANG*WFLTA1+SRSEP2	TVFU117
000118	001	W(4)=RANG*WFLTA2-SRSEP2	TVFU118
000119	001	U(5)=1.	TVFU119
000120	001	C	TVFU120
000121	001	120 T1=FUNC(0.+SRSEPI+FANG+HANG+ANGLES+NVIKON)	TVFU121
000122	001	U(5)=.5*(L1)/(C(.6)+T1)	TVFU122
000123	001	T1=FUNC(-ANGLEFR+SRSEPI1+EANG+RANG+ANGLES+NVIKON)	TVFU123
000124	001	U(7)=.5*(L1)/(C(.7)+T1)	TVFU124
000125	001	IF (U(7).LT.0.) GO TO 130	TVFU125
000126	001	W(1)=RANG*(WFLTA2-SRSEP2)	TVFU126
000127	001	GO TO 140	TVFU127
000128	001	C	TVFU128
000129	001	130 W(1)=FHINC(0(.6)+U(7)+RANG+ANGLE+EANG+SRSEPI)	TVFU129
000130	001	140 U(2)=VFHINC(DELTA1+HANG+DF+SRSEPI)	TVFU130
000131	001	T4=TAN(ANGLE+EANG)	TVFU131
000132	001	W(3)=FHINC(T4+HANG+DF+SRSEPI)	TVFU132
000133	001	W(5)=FHINC(DELTA2+HANG+DF+SRSEPI)	TVFU133
000134	001	T4=TAN(ANGLE-EANG)	TVFU134
000135	001	W(6)=FHINC(T4+HANG+DF+SRSEPI)	TVFU135
000136	001	CALI FNK (1+3)	TVFU136
000137	001	U(8)=U(3)	TVFU137
000138	001	U(9)=U(3)	TVFU138
000139	001	CALI FNK (4+7)	TVFU139
000140	001	U(10)=U(7)	TVFU140
000141	001	U(11)=U(7)	TVFU141
000142	001	U(12)=U(8)+U(10)	TVFU142
000143	001	IF (U(12).LT.0.) U(12)=0.	TVFU143
000144	001	U(11)=EANG	TVFU144
000145	001	C	TVFU145
000146	001	150 CONTINUE	TVFU146
000147	001	IF (IFLAG.FQ.1) GO TO 160	TVFU147
000148	001	IF (ICAL.FQ.1) GO TO 230	TVFU148
000149	001	IF (U(12).LE.0.) GO TO 230	TVFU149
000150	001	IFLAG2=1	TVFU150
000151	001	GO TO 230	TVFU151
000152	001	C	TVFU152
000153	001	160 IF (ARS(C(4)).GE..05/A(1+1)) GO TO 170	TVFU153
000154	001	IFLAG2=0	TVFU154
000155	001	GO TO 230	TVFU155
000156	001	C	TVFU156
000157	001	170 IF (ICAL=2) 230+140+200	TVFU157
000158	001	180 IF (U(12).GT.0.) GO TO 190	TVFU158
000159	001	IFLAG2=1	TVFU159
000160	001	C(4)=.5*ARS(C(4))	TVFU160
000161	001	GO TO 230	TVFU161
000162	001	C	TVFU162
000163	001	190 IF (IFLAG2.EQ.0) GO TO 230	TVFU163
000164	001	C(4)=.5*ARS(C(4))	TVFU164
000165	001	GO TO 230	TVFU165
000166	001	C	TVFU166
000167	001	200 IF (U(12).GT.0.U(12)) GO TO 230	TVFU167
000168	001	IF (U(12).LT.0.) GO TO 220	TVFU168
000169	001	IF (HANG=.1.5*ARS(C(4)).GT.HANG1) GO TO 210	TVFU169

U00170	U01	HANGERANG=ARS(C(4))	TVFU170
U00171	U01	C(4)=.5*ARS(C(4))	TVFU171
U00172	U01	GO TO 230	TVFU172
U00173	U01	C	TVFU173
U00174	U01	210 RANGERANG=2.*ARS(C(4))	TVFU174
U00175	U01	C(4)=.5*ARS(C(4))	TVFU175
U00176	U01	GO TO 230	TVFU176
U00177	U01	C	TVFU177
U00178	U01	220 C(4)=-.5*C(4)	TVFU178
U00179	U01	230 DO 240 I1=1,13	TVFU179
U00180	U01	U(I1)=U(I1)	TVFU180
U00181	U01	240 CONTINUE	TVFU181
U00182	U01	IF (IFLAG,0,0) GO TO 250	TVFU182
U00183	U01	HANGERANG+C(4)	TVFU183
U00184	U01	GO TO 40	TVFU184
U00185	U01	C	TVFU185
U00186	U01	250 RETURN	TVFU186
U00187	U01	END	TVFU187

U00001	U01	FUNCTION UFUNC(A,RAN,DFIELD,SRSEP)	UFUN 1
U00002	U01	UFUNC=A*(RAN-(SRSEP/(2.*A))-DFIELD)	UFUN 2
U00003	U01	RETURN	UFUN 3
U00004	U01	END	UFUN 4-

U00001	U01	SUBROUTINE UPDATE (ISTART)	UPDA 1
U00002	U01	C	UPDA 2
U00003	U01	C	UPDA 3
U00004	U01	C	UPDA 4
U00005	U01	COMMON /IOLIST/ NIVRON, ISTPF, INTPE, WAVEHT, UTMRUF, TARGLY,	UPDA 5
U00006	U01	1, TADGHT, SRSEP, SPWR, SULFL, RUEFL, OPTRAN, FNUM, HFIGHT,	UPDA 6
U00007	U01	2, RANGMX, RANGE, HANGAL, HANGER, HANGMX, SWTHL, SWTHR, SWTHR,	UPDA 7
U00008	U01	3, SWTHL, AVSWL, AVSHR, AVSW, LINCEN, LINEUG, CTRCEN, CTHEDG,	UPDA 8
U00009	U01	4, NUFAUL, NUEXWH, LIMFLG	UPDA 9
U00010	U01	C	UPDA 10
U00011	U01	C	UPDA 11
U00012	U01	COMMON /UPD/ STEP(14), EDVAL(14)	UPDA 12
U00013	U01	C	UPDA 13
U00014	U01	DIMENSION VALUE(14)	UPDA 14
U00015	U01	DIMENSION HEG(14), ID(14)	UPDA 15
U00016	U01	DATA NTRY /0/	UPDA 16
U00017	U01	C	UPDA 17
U00018	U01	EQUIVALENCE (VALUE(8),WAVEHT)	UPDA 18
U00019	U01	C	UPDA 19
U00020	U01	VALUE(1)=NIVRON	UPDA 20
U00021	U01	VALUE(2)=ISTPF	UPDA 21
U00022	U01	VALUE(3)=INTPE	UPDA 22
U00023	U01	IF (NTRY,NE,0) GO TO 20	UPDA 23
U00024	U01	NUM=20	UPDA 24
U00025	U01	DO 10 I2=1,8	UPDA 25
U00026	U01	IF (ABS(STEP(I2))-EA,0,0) GO TO 10	UPDA 26
U00027	U01	NUM=NUM+1	UPDA 27
U00028	U01	ID(I,NUM)=1	UPDA 28
U00029	U01	BEG(NNUM)=VALUE(I)	UPDA 29
U00030	U01	10 CONTINUE	UPDA 30
U00031	U01	NTRY=1	UPDA 31
U00032	U01	C	UPDA 32
U00033	U01	20 IF (NNUM,NE,0) GO TO 70	UPDA 33
U00034	U01	I=1	UPDA 34
U00035	U01	30 IN=ID(I)	UPDA 35
U00036	U01	IF (VAL,E(IN),NE,EDVAL(IN)) GO TO 50	UPDA 36
U00037	U01	IF (I,NE,NUM) GO TO 40	UPDA 37
U00038	U01	NTRY=0	UPDA 38
U00039	U01	GO TO 70	UPDA 39
U00040	U01	40 VALUE(IN)=HEG(I)	UPDA 40
U00041	U01	I=I+1	UPDA 41
U00042	U01	GO TO 30	UPDA 42
U00043	U01	C	UPDA 43
U00044	U01	50 ISTART=1	UPDA 44
U00045	U01	VALUE(IN)=VALUE(IN)+STEP(IN)	UPDA 45
U00046	U01	IF (ABS(STEP(IN)),EA,STEP(IN)) GO TO 60	UPDA 46
U00047	U01	IF (VAL,E(IN),LT,EDVAL(IN)) VALUE(IN)=EDVAL(IN)	UPDA 47
U00048	U01	GO TO 70	UPDA 48
U00049	U01	60 IF (VAL,E(IN),GT,EDVAL(IN)) VALUE(IN)=EDVAL(IN)	UPDA 49
U00050	U01	C	UPDA 50
U00051	U01	70 NIVRON=VALUE(1)	UPDA 51
U00052	U01	ISTPF=VALUE(2)	UPDA 52
U00053	U01	INTPE=VALUE(3)	UPDA 53
U00054	U01	C	UPDA 54
U00055	U01	RETURN	UPDA 55
U00056	U01	C	UPDA 56
U00057	U01	END	UPDA 57-

U00001	U01	FUNCTION VFUNC(A,RAN,DFIELD,SRSEH)	VFUN 1
U00002	U01	CALCULATE SOURCE R. AND RECEIVER L WIDTH	VFUN 2
U00003	U01	VFUNC=A+(RAN+(SRSEH/(2.*A))-DFIELD)	VFUN 3
U00004	U01	RETURN	VFUN 4
U00005	U01	END	VFUN 5-
U00001	U01	FUNCTION XFUNC(THET,SRSEH,LANG,RAN,ANG,T)	XFUN 1
U00002	U01	C	XFUN 2
U00003	U01	COMMON /IVCOM/ A(2,2), B(2,11,2), C(20), D(2,2), E(10),	XFUN 3
U00004	U01	COMMON /IVCOM/ A(2,2), B(2,11,2), C(20), D(2,2), E(10),	XFUN 4
U00005	U01	1 FA(2,2), 6(2,2,3), H(2,2,3), N(10), O(15), P(2,2),	XFUN 5
U00006	U01	2 Q(15), S(2), T(11), W(10), Z(2), PII	XFUN 6
U00007	U01	C	XFUN 7
U00008	U01	F1=(SRSEH/(TAN(EANG+ANG)+TAN(EANG-THET)))	XFUN 8
U00009	U01	F2=cos(THET-EANG)	XFUN 9
U00010	U01	F3=(O(1,1)*A(1,1))+(O(1,2)*A(1,2))/F2	XFUN 10
U00011	U01	F4=(A(FINC(F3*F1))/F1-(A(FINC(F3*HAN))/RAN))	XFUN 11
U00012	U01	XFUNC=1/2*(COS(THET)*O(1,1)*F2*F4)	XFUN 12
U00013	U01	RETURN	XFUN 13
U00014	U01	END	XFUN 14-

BEST AVAILABLE COPY